REPORT DOCUMENTATION PAGE

AFRL-SR-AR-TR-05-

0125

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
30 March 2005	Final Project Report	1 January 2002 - 31 December 2004

4. TITLE AND SUBTITLE
Wireless Multiple Access Communications Using Collision Frequency Shift Keying

5a. CONTRACT NUMBER
F496200210249

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Pérez, Lance C.
Chen, Xia

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Department of Electrical Engineering
University of Nebraska, Lincoln
Lincoln, NE 68588-0511

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research/Office of Naval Research

NM

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR/)NR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
N/A

14. ABSTRACT

This report investigates the design and performance of a novel wireless multiple access scheme based on collision frequency shift keying (CFSK). The CFSK system has been shown to have a greater capacity than a comparable CDMA system. However, the optimal multiuser receiver for the CFSK system has complexity that is exponential in the number of users. Several suboptimal reduced complexity receivers for the CFSK system are developed and their performance evaluated via computer simulation. These suboptimal receivers are shown via computer simulation to have performance close to that of the optimal receiver. These results indicate that CFSK is a promising system for ad hoc communications networks that cannot use a highly coordinated system.

15. SUBJECT TERMS
wireless communications, iterative receivers

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Lance C. Pérez
U	U	U	SAR	142	19b. TELEPHONE NUMBER *(Include area code)* 402-472-6258

Wireless Multiple Access Communications Using Collision Frequency Shift Keying

AFOSR Grant F496200210249

Principal Investigator: Lance C. Pérez, Ph.D.

Executive Summary

Objectives:
A new multiple access method, called Collision Frequency Shift Keying (CFSK), is investigated for wireless communication systems. The specific objectives of this work are to:

- Develop a reduced complexity iterative receiver for CFSK systems.
- Simulate the performance of highly loaded CFSK communication systems using iterative receivers.
- Perform thorough performance comparisons of CFSK and CDMA systems.
- Design and test detectors suitable for analog implementations of iterative CFSK receivers.

Accomplishments:
The theoretical capacity of the multiuser CFSK system is well known to come at the expense of an optimal multiuser receiver whose complexity is exponential in the number of the users. In fact, we conjecture that the optimal detection problem is provably NP complete. In order to achieve the theoretical promise of the CFSK system and to improve on the capacity of a comparable CDMA system a reduced complexity, suboptimal receiver is necessary. We have developed two novel iterative receivers.

The iterative receivers are modifications of the maximum a posteriori (MAP) based consensus decoder. The metric of the MAP algorithm in the noisy CFSK channel is, in general, too computationally complex in a highly loaded environment. To reduce this complexity to a usable level, we have developed the simplified K-Most Probable Combination (KMPC) and Two-Most Probable Combination (TMP) metrics. Both of these metrics substantially reduce the complexity of the overall receiver. Simulation results have shown that this metric results in a modest loss of performance compared to the optimal receiver.

Simulation results show that the CFSK system with our iterative receivers outperform conventional CDMA systems. These results strongly indicate that CFSK is a promising system for ad hoc communications networks where a highly coordinated system like CDMA is not practical. The iterative receivers utilize MAP decoders that have been shown to have viable low power, high speed analog implementations.

Personnel Supported:
Lance C. Pérez, PI.
Xia Chen, Research Assistant, Ph.D. expected December 2005(fully supported).
Eric Psota, Research Assistant, M.S.E.E. expected May 2006 (partially supported).
Fan Jiang, Research Assistant, Ph.D. expected December 2005 (partially supported).

Journal Publications:

D. Leon, S. Balkir, M. Hoffman and L. C. Pérez, "Pseudo-Chaotic PN-sequence generator circuits for spread spectrum communications", *IEE Proceedings-Circuits, Devices and Systems*, vol. 151, no. 6, pp. 543-550, December 2004.

X. Chen, L. C. Pérez and C. B. Schlegel, "Performance of a CFSK Wireless Communication System with Iterative Detection", in preparation for the *IEEE Transactions on Communications*.

Referred Conference Proceedings:

X. Chen and L. C. Pérez, "Iterative Detection of a CFSK System", presented at the Special Session on Wireless Communications at the 2003 Richard Tapia Conference, Atlanta, Georgia, October 15-18. 2003.

X. Chen and L. C. Pérez, "A CFSK System with Iterative Detection", presented at the 2004 IEEE GLOBECOM Conference, Dallas, TX, November 29- December 3, 2004.

X. Chen and L. C. Pérez, "Non-Binary Trellis Codes for a CFSK System with Intensity Information", submitted to the 2005 International Symposium on Information Theory, Adelaide, AUSTRALIA.

Interactions:

Member, Technical Program Committee, 2004 IEEE Globecom Conference
Member, Technical Program Committee, 2005 International Symposium on Information Theory

Wireless Multiple Access Communication

Using Collision Frequency Shift Keying

Final Project Report

AFOSR Grant F496200210249

Principal Investigator: Lance C. Pérez, Ph.D.

by

Chen Xia and Lance C. Pérez

Department of Electrical Engineering

209N Walter Scott Engineering Center

University of Nebraska, Lincoln

Lincoln, NE 68588-0511

Contents

List of Figures

List of Symbols and Abbreviations

AWGN	Additive white Gaussian noise
CDMA	Code division multiple access
CFSK	Collision frequency shift-keyed
DS-SSMA	Direct sequence spreading spectrum multiple access
FDMA	Frequency division multiple access
FH-SSMA	Frequency hop spreading spectrum multiple access
GSM	Globe system for mobile communication
IC	Interference Cancellation
ID	Iterative detection
KMPC	K most probable combinations
kMPF	k most probable frequencies
KS	Kautz Singleton code

MAP	Maximum *a posteriori*
MLSD	Most likelihood sequence detection
NSMP	Narrow sense most probable combination
P.D.F.	Probability density function
P.S.D.	Power spectral density
RSC	Recursive systematic convolutional code
RS	Reed Solomon
SNR	Signal to noise ratio
SSMA	Spreading spectrum multiple access
TC	Turbo Code
TCMA	Trellis-coded multiple access
TDMA	Time division multiple access
UDBC	Uniquely decodable block code
UDC	Uniquely decodable code
UUF	Unique user feature
D_j	Binary difference matrix in the j^{th} iteration in the construction of UDBC
$\tilde{D}$	Binary difference matrix for the construction of binary UDBC

$\tilde{D}^*$	Nonbinary difference matrix for the construction of nonbinary UDBC
K	Number of combinations used in KMPC or kMPF
N_o	Noise power defined as one-sided power spectrum
N	Length of UDBC codeword
M	Number of frequencies
T	Number of users
U_i	The i^{th} user that access to the system
f_j	The j^{th} frequency
$\bar{F}$	Received frequency block consisting of frequency vectors from time 0 to $T-1$
ρ_j	Received energy for frequency j
μ_j	Multiplicity factor of frequency j
θ	Phase difference between transmitter and receiver
E_s	Symbol energy
$p_{j_u}^{(u)}$	A priori probability for frequency f_{j_u} for user U_u
$P^{(u)}$	A priori probability block for all frequencies of user U_u
j	Index, counting the frequencies, range: $0 \cdots M-1$
u	Index, counting the users, range: $0 \cdots T-1$

μ_j	Multiplicity of frequency f_j, denotes the number of transmissions on f_j
α	Forward metric of MAP algorithm
β	Backward metric of MAP algorithm
γ	Branch transition probability for a symbol
Θ_j	Synchronous phase for transmitting frequency f_j
$\Phi_{j_u}^{(u)}$	Asynchronous phase for transmitting frequency f_{j_u} of user U_u

Chapter 1

Introduction

In the last several decades, researchers have witnessed rapid developments and widespread applications in the area of multiple access communication systems, especially of mobile communication systems. In the first generation of mobile communication systems, analog transmission using frequency division multiple access (FDMA) [1] was used. In such systems, each user occupies a separate frequency band for transmission. When a large number of users access the system, a huge amount of frequency resources are required. In other wireless multiple access communication systems, such as the European recommendation for Globe Spéciale Mobile (GSM) communication system [2], time division multiple access (TDMA) is used. For TDMA, all users share the same frequency band by dividing the channel into small time slots and assigning the time slots to different users. However, both

FDMA and TDMA are not efficient in utilizing the channel resources. Huge bandwidth is required for both systems when a lot of users access them.

Spread spectrum communication originated in military applications in the 1950s. Through the appearance of the famous IS95 standard proposed by Qualcomm [3][4][5], more people realized that spread spectrum multiple access (SSMA) is much more efficient than TDMA and FDMA. SSMA is also known as code division multiple access (CDMA) [6], where pseudonoise (PN) sequences are used to transform the signals from other users into a Gaussian like noise signal. The PN sequence can be applied in the time domain, which is called direct-sequence SSMA (DS-SSMA), or in the frequency domain, which is called frequency-hop SSMA (FH-SSMA). Each user in these systems is uniquely identified by their PN code. Thus, the signals are hopping in the time or frequency domain with a hopping pattern controlled by the PN code. The conventional detection method for the DS- and FH-SSMA systems is not optimal in the sense that it implements single user detection by treating other users as purely unwanted interference. Optimal detectors require multiuser detection techniques that treat other users' information as useful in detecting a user's information sequence. A variety of multiuser detection methods for DS-SSMA systems [7][8][9][10] and for FH-SSMA systems [11][12] have been proposed in resent years.

The third category of multiple access systems is based on collision-type channels. Network communications based on the ALOHA protocol [13] is a famous example of the collision-type channel. Communications on collision type channels is based on the avoidance of collisions. It is not difficult to show that the ALOHA system is even less efficient than the TDMA system. However, resent research on trellis-coded multiple access (TCMA) gives us a new approach for communication on collision-type channels [14]. For TCMA, all users use trellis-coded modulation schemes and share a common signal space for the modulation. Thus, when different users transmit on the same signal a collision occurs. In [15], Brännström showed that the TCMA channel is more efficient than the CDMA system.

This research project considered a new multiple access communication system with multiuser detection called the collision frequency shift-keyed (CFSK) system. The CFSK channel is a kind of collision type multiple access channel [16] where different users share the common channel simultaneously and thus collisions occur. For each user, M-ary frequency shift-keyed (FSK) [6] modulation is used and, since all users share the same set of frequencies, collisions happen when different users transmit the same frequency at the same time. Hence, this is different from an FDMA system where each user is assigned a subband of the whole spectrum and no collisions happen on

the separated subbands. In a CFSK system, users send their signals within the same wideband spectrum, which is similar to the CDMA system where each user's channel signal will span the whole wideband spectrum. However, in the CFSK system one user only transmits a single frequency point in a symbol period. In a CDMA system, a signal always occupies the whole spectrum.

There are some similarities and differences between the CFSK system and a FH-SSMA system. In both systems, the users may transmit the same frequency simultaneously and a collision happens. Also, the transmitted frequencies in different symbol periods are changing in a random-like pattern. In the CFSK system, the frequency hopping pattern is controlled by each user's channel code, and thus controlled by each user's information sequence. On the other hand, in an FH-SSMA system the frequency hopping code is given by a PN sequence that is not dependent on the user's information sequence.

Chang and Wolf first derived the channel capacity for the noiseless T-user M-frequency multiple-access channel with and without intensity information [17] in 1982. In [18], Grant and Schlegel compared the results of Chang and Wolf and the capacity of the CDMA system given by [19][20][21]. The results show that the capacity for the CFSK channel with intensity information has

a higher sum channel capacity than the conventional and multiuser CDMA systems, especially when the spreading factor is of moderate values. In Figure 1.1, the sum capacity for the synchronous CFSK system and CDMA system are shown. When the spreading factor is of moderate or high values, the sum capacities of the CFSK channel with or without intensity information are almost 4 times the sum capacity of the CDMA system with single user or multiuser detection. When the spreading factor is lower than one, the CFSK channel with intensity information still has a higher sum capacity than that of the CDMA system with multiuser detection. However, the sum capacity for the CFSK channel without intensity information decreases drastically when the spreading factor is lower than 1. The study of the CFSK system with intensity information when the spreading factor is of moderate values is of considerable interest. This is the emphasis of the work in this project.

For asynchronous CFSK channels, the spectral efficiency decreases, especially when the number of users is larger than the number of frequencies, i.e., the spreading factor is lower than 1. The performance of the asynchronous CFSK system is discussed in detail in Chapter 3.

There are other advantages of the CFSK system over conventional multiple access systems which provide additional motivation for investigating the CFSK system. First, it is very resistant to the near-far effect, as the re-

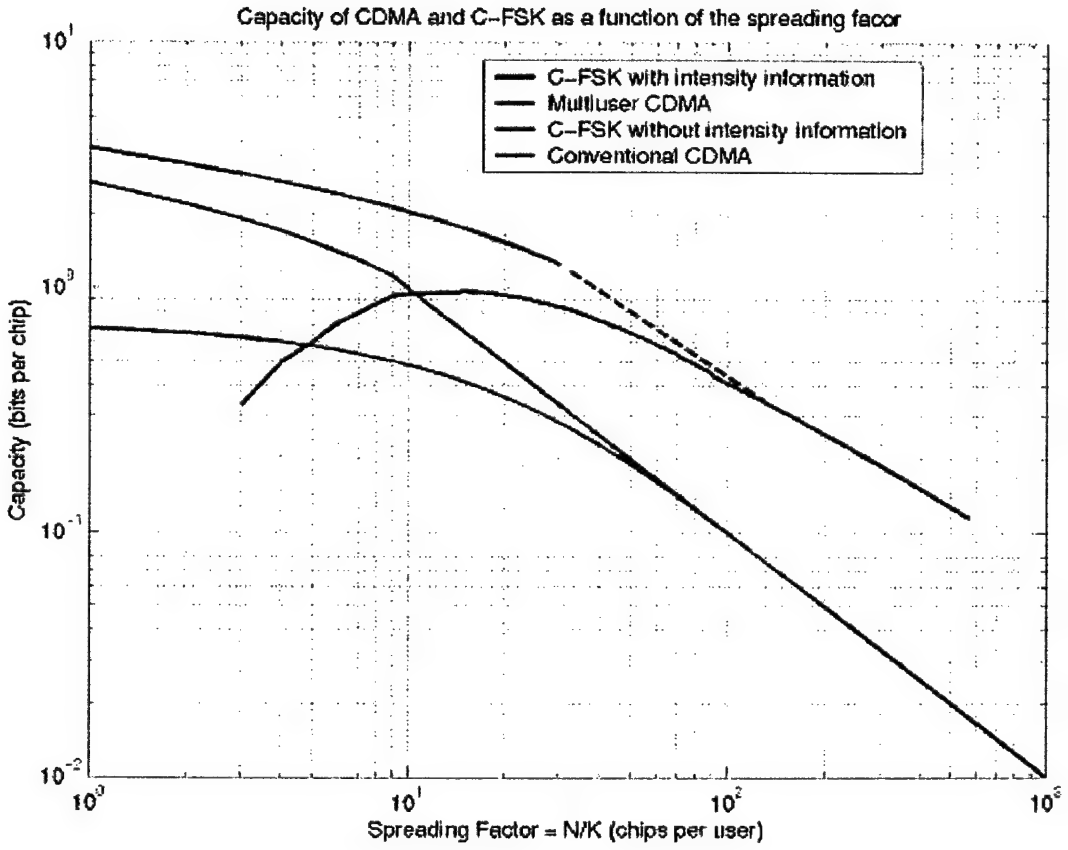

Figure 1.1: A comparison of the sum capacity of CDMA and CFSK systems.

ceiver utilizes multiuser detection. Verdú proved that multiuser detection can provide good resistance to near-far effect in [7]. Second, inter-symbol interference (ISI) is no longer a problem in the CFSK system since aliasing in the frequency domain is of more significance than aliasing in time domain. Third, because of the frequency diversity of the CFSK system, its performance on frequency selective fading channels is better than that of conventional multiple access system. Finally, the transmitter of the CFSK system is of very low complexity.

The channel metric for the synchronous and asynchronous CFSK channel with intensity information is given by Arnold, Schlegel and Pérez [22] in 1996. Based on the channel metric, an optimal receiver can be constructed using the maximum-likelihood sequence detection (MLSD) method [7]. Aulin and Espineira discussed MLSD for the TCMA system in [14]. The CFSK system can in fact be viewed as a multi-dimensional version of a TCMA system where each frequency spans a dimension of the signal space. The MLSD method constructs a super-code and a super-trellis by considering all users' memories and inputs together, and does maximum-likelihood detection on the super-trellis. When the number of users gets large, the states of the super-trellis increase exponentially and the MLSD method is impossible. Thus, a suboptimal method using iterative detection is preferred. This is

another topic addressed in detail in this project.

This report begins with a description of the CFSK system in Chapter 2. The spectral efficiency of the synchronous and asynchronous CFSK channels is discussed in Chapter 3, which provides a theoretical foundation for later discussions. Several iterative multiuser detection techniques are discussed in Chapter 4. The performance of the iterative dectectors is discussed in Chapter 5. These results show that decent performance can be achieved for the system with low and moderate loads. In Chapter 6, the issue of code design for the CFSK system is considered. Finally, conclusions and recommendations for further research are given in Chapter 7.

Chapter 2

CFSK System Description

2.1 Overview

A multiple-access multiuser system based on the Collision Frequency Shift-keyed (CFSK) channel is described in Figure 2.1. Suppose that there are T users access to the system. Each user generates and transmits the signals independently. The transmitter includes a channel encoder, a channel interleaver and an M-ary FSK modulator. The channel code and the interleaver are optional. For the M-ary FSK modulation, all users share the same set of frequencies with size M. Since there is no cooperation among the T users when they generate and send the signals, it is possible that some of the users send the same frequency at the same time, and collision happens. At the receiver, the demodulator operators on the signals over the M frequencies. If the demodulator outputs the number of users that transmit on each fre-

quency by detecting the energy level on that frequency point, the channel is called the CFSK channel with intensity information. In this case, the output of the demodulator is given by a vector $\bar{F}$, in which the entries denote the number of transmissions on the frequencies. On the other hand, when the demodulator only detects whether there is any transmission on each frequency point, the channel is called the CFSK channel without intensity information. In this case, the entry of the vector $\bar{F}$ is one if there are transmissions on the corresponding frequency, zero otherwise. The decoder does multiuser detection on the output of the demodulator, $\bar{F}$. For each user, the component decoder regards other users' signals as useful information and makes use of them. The output of the multiuser detection is the information sequence for each user.

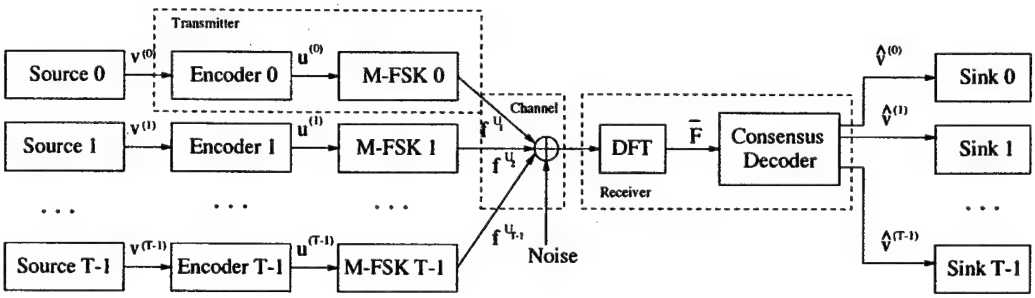

Figure 2.1: The CFSK system.

2.2 The CFSK Transmitter

2.2.1 Overview

Each user's transmitter contains a channel encoder, a random channel interleaver and an M-ary FSK modulator. The channel encoder and interleaver are optional. The code can be block code or convolutional code. The purpose for the channel code is to provide error control for the system. Moreover, the application of the channel code can provide the user unique feature (UUF) [23] so that the users can be distinguished at the receiver. When the trellis code presents in the system, the CFSK system can be regarded as a trellis coded modulation system. The random channel interleaver after the channel encoder is also optional. The interleaver is symbol-based. It can also provide the UUF. Purpose for the channel interleaver is to improve the system performance, which will be discussed in Chapter 4. In the M-ary FSK modulation, the user's coded sequence is mapped to the frequency signals. Considering that there are M frequencies, the signal space is an M-dimensional space, where each dimension denotes a frequency. Application of different signal mappings can also provide UUF.

2.2.2 User Unique Feature (UUF)

The conception of UUF is given by [23]. In the multiple access system, all users share the common channel. Thus, the users need some particular properties to distinguish themselves from others. For example, in the SSMA systems, each user has a specific spreading sequence. In a collision type system, such as CFSK, there are many ways to achieve UUF. As described previously, UUF can be achieved by using different channel codes for different users. Or, it can be achieved by using different channel interleavers or different signal mappings for different users. And, these schemes can be combined together to achieve better performance. Different UUF schemes require different decoder structures for the multiuser detection.

In this project, we focus on the using different channel codes for different users to achieve the UUF. Trellis codes, especially convolutional codes, are considered. All users are using the same signal mapping over a set of M frequencies. Assume that (n, k) convolutional codes are used for all the users. Then, we have $M = 2^n$, i.e. each n bits of coded sequence are mapped to a frequency. The channel interleaver is optional in this project, which will be applied in discussion of the performance for the iterative detection in Chapter 5.

2.2.3 Unique Decodability

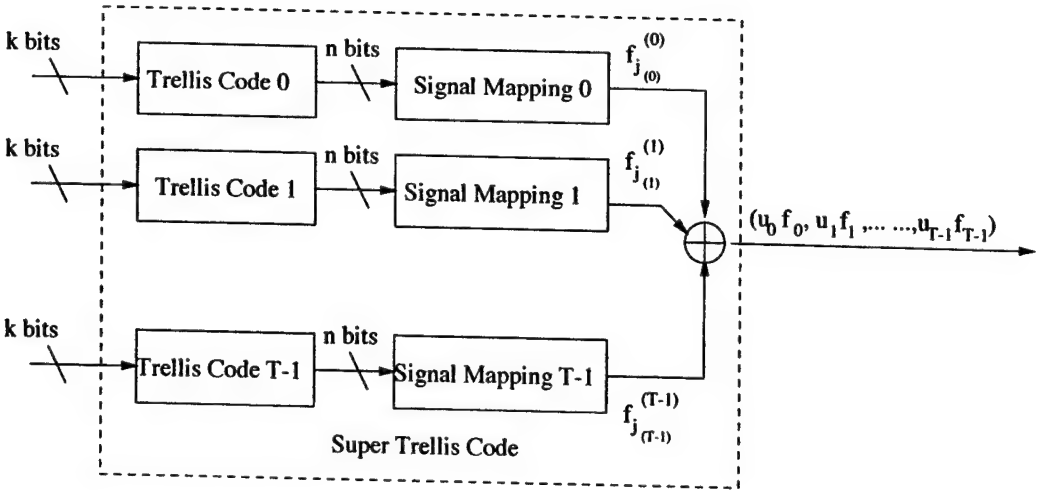

Figure 2.2: The super trellis code of a T-user CFSK system.

The transmitter of each user is considered as a trellis coded modulation (TCM) system, since both block codes and convolutional codes are trellis codes. When all users are considered together, the transmitter can be regarded as a super trellis code. In Figure 2.2, the super trellis code is described. The code is constructed by aligning all users' input bits together, and aligning all the memories together. When rate k/n, memory m convolutional codes are applied, the trellis code of the TCM has kT bits input and mT memories (state number is 2^{mT}), where T is the number of users. Since each n bits of coded sequence are mapped to a frequency and there are M frequencies, the code symbols on the branch of the super trellis can be expressed by the M-ary vector $\bar{F}$. The j^{th} entry of the $\bar{F}$ denotes the number

of users that transmit the j^{th} frequency. The code symbols described by $\bar{F}$ occupy the points in an M-dimensional signal space. For each signal point, the sum of its coordinates equals to the user number T. As an example, signal space for a 3-user 3-frequency system can be described in Figure 2.3.

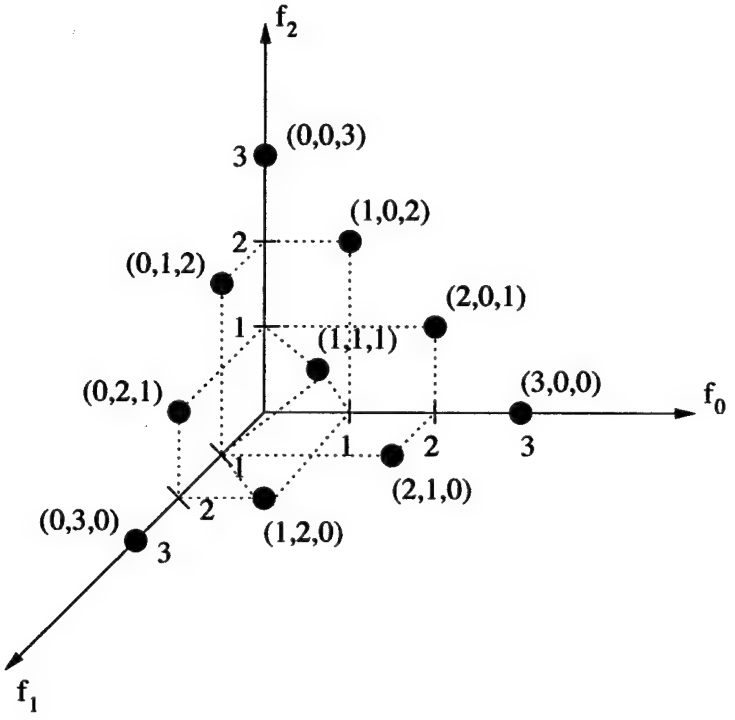

Figure 2.3: The signal space for a 3-user 3-frequency CFSK system.

Unique decodability of the super trellis code has significant influence on the performance of the CFSK system. The definition of the uniquely decodable code (UDC) for a multiple access channel was first given by Kasami and Lin in [24] in 1976. Their discussion is based on the block code for the binary

adder channel. In 1979, Chang and Weldon gave some construction methods for the uniquely decodable block codes for the T-user adder channel. These constructions can approach the maximal code rates given by capacity of the T-user adder channel. Further, Chang and Wolf put forward the definition of the uniquely decodable block code for the CFSK channel and gave the construction methods for the codes, which can approach the channel capacity. The conception of the uniquely decodable convolution code is first given by Peterson in 1979 in [25], where the two-user binary adder channel is assumed. Chevillat developed the results further in [26]. He gave the definition and some constructions for the uniquely decodable trellis code for the N-user adder channel. For the CFSK system, if every path through the super-trellis has a distinct code sequence, then the code set is uniquely decodable. In this situation, each code path has to start from the state zero and end at the state zero. Each codeword in the code sequence is given by $\bar{F}$. Brute force checking shows that the two codes given in Figure 2.4 is uniquely decodable.

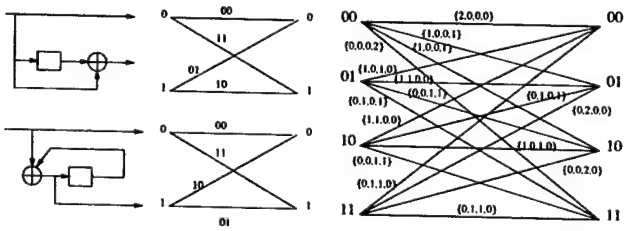

Figure 2.4: A two-user super-trellis.

It has been shown that to construct a set of uniquely decodable codes with error control for a large number of users is extremely difficult [27][28]. For later discussion, the recursive systematic convolutional codes (RSC) are generated randomly for the users. An important reason for applying RSC is that the systematic convolutional codes with canonical controller form are proved to be non-catastrophic [29][30] . Since RSC is of such form, the randomly generated codes are non-catastrophic automatically. Also, a couple of generation rules [22] can guarantee the uniform implementation for all users.

In Figure 2.5, the code set for the 3-user system is not uniquely decodable. For instance, the three state sequences through the trellises are

$$\begin{aligned} code(a) &: 00, 00, 10, 01, 00, 00, 10, 11, 01, 00, \\ code(b) &: 00, 00, 00, 10, 11, 01, 10, 11, 01, 00, \\ code(c) &: 00, 10, 01, 10, 11, 01, 10, 01, 00, 00. \end{aligned} \tag{2.1}$$

The corresponding output sequence on the super-trellis is

$$\begin{aligned} &(2,0,1,0),(1,0,1,1),(1,1,0,1),(1,1,1,0),(1,2,0,0), \\ &(1,1,0,1),(0,1,1,1),(0,1,1,1),(1,0,1,1). \end{aligned} \tag{2.2}$$

There is another set of state sequences through the trellises:

$$\begin{aligned} code(a) &: 00, 00, 10, 11, 11, 11, 11, 11, 01, 00, \\ code(b) &: 00, 00, 00, 00, 00, 00, 10, 01, 00, 00, \\ code(c) &: 00, 10, 01, 10, 01, 10, 11, 11, 01, 00, \end{aligned} \tag{2.3}$$

with the same output sequence given in (2.2). Hence, whenever the code sequence in (2.2) is received, one of paths in (2.1) and (2.3) has to be decided randomly. So, it is always possible that decoding error will happen. And, this will lead to an error floor for the bit error performance of the CFSK system.

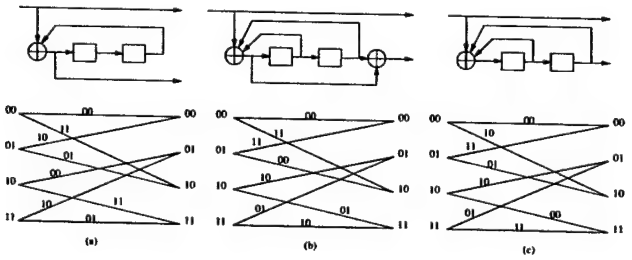

Figure 2.5: An code set for the 3-user CFSK system, which is not uniquely decodable.

The effects of the non-uniquely decodable codes will be discussed in Chapter 5. In Chapter 6, some constructions of uniquely decodable codes will be discussed.

2.3 The CFSK Channel

2.3.1 Overview

For the M-ary FSK modulation, let the set of M frequencies be

$$\mathcal{S} = \{f_0, f_1, \ldots, f_{M-1}\}. \tag{2.4}$$

Then, the output signal from the channel has the form

$$s(t) = \sum_{j=0}^{M-1} \left[\phi(\mu_j)\sqrt{E}cos(2\pi f_j t + \theta_j) + n_j\right], \tag{2.5}$$

where μ_j is the multiplicity factor that denotes the number of users that transmit frequency f_j, E is the signal energy. $\phi(\mu_j)$ is a function of μ_j that depends on the synchronization of the users, which will be discussed later. θ_j is a uniformly distributed the phase for frequency f_j. n_j is the additive noise. In Chapter 3, specific math model for the CFSK will be given for the synchronous and asynchronous channels.

2.3.2 Synchronization

In Figure 2.6, three synchronization modes are described. For the full synchronous mode, all users start transmission at the same time and with the same phase for each frequency. For the "asynchronous over time step" mode, the users may start transmission at different time. But, they transmit a

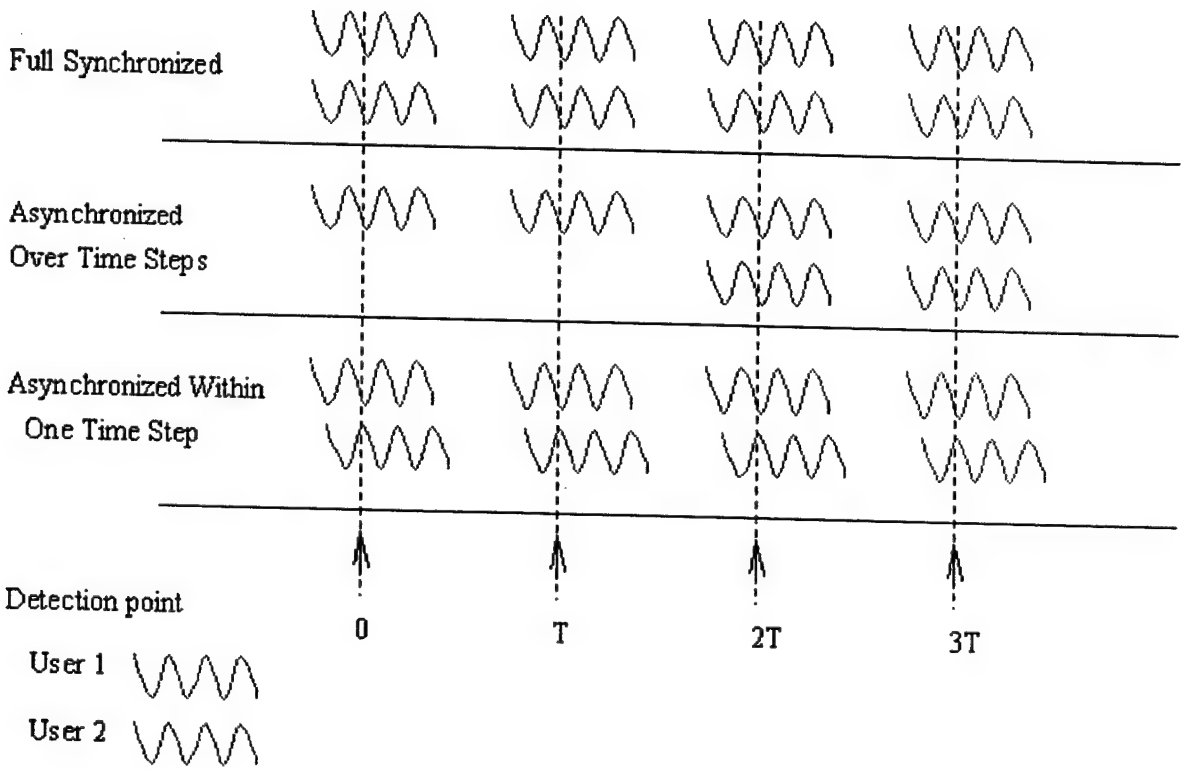

Figure 2.6: Synchronization modes on the CFSK channel.

frequency with the same phase. Thus, the symbols are synchronized. If additional communication protocol can guarantee that the receiver knows when each user starts transmission, then the above two modes are the same for the receiver. Thus, an assumption for later discussion is that the the above two modes are all synchronous case. The third case in Figure 2.6 is the asynchronous mode. In this mode, the users transmit their symbols with different phases no matter they start transmission together or not. In (2.5), the function $\phi(\mu_j)$ depends on the synchronization mode. For the synchronous mode, $\phi(\mu_j) = \mu_j$. This is because that all users transmit f_j with the same phase, which is equal to θ_j, thus the result signal on f_j is just a sum of all the signals of the users. For the asynchronous case, the value of $\phi(\mu_j)$ is a random variable with variance $\mu_j E/2$, which will be discussed in Chapter 3.

2.4 CFSK Receiver Structures

2.4.1 Overview

The frequency spectrum of the channel output signals only occupies on single frequency points. Thus, a simple demodulation method can be achieved by using the discrete Fourier transformation (DFT). Since M frequencies are used in the FSK modulation, the signal has a bandwidth of M. So, $2M$ points need to be sampled in each symbol period. The sampled points are

feed into the $2M$-point DFT. Then, the energy level and phase for each frequency are output from the DFT function. For frequency f_j, suppose that (2.5) gives the channel signal. Then, the DFT output on f_j can be given by a phasor expression

$$s_j = \rho_j \angle \theta_j = [\phi(\mu_j)]^2 E \angle \theta_j. \tag{2.6}$$

For synchronous case, ρ_j equals to $\mu_j^2 E$ without considering the noise. For asynchronous case, ρ_j is a random variable with expectation $\mu_j E$. Thus, the output of DFT over all frequencies can be expressed by the vector

$$\bar{F} = (\rho_0, \rho_1, \ldots, \rho_{M-1}). \tag{2.7}$$

The multiuser detection takes the vector $\bar{F}$ as input. For each user, the information from other users is not treated as interference. The information from other users are applied in decoding each user's information. Generally, there are two specific approaches for multiuser detection.

2.4.2 Optimal Detection of CFSK

The optimal detection gives the best performance for the multiuser detection for the CFSK system. It operates on the super trellis of the system. By maximum likelihood sequence detection (MLSD) on the super trellis [14], the decoder can provide the best estimation for the code path through the super trellis. From a specific state path through the super trellis, each user's

state path through his trellis can be uniquely decided. Thus, each user's coded sequence and information sequence can be decided. Note that when the code set is not uniquely decodable, there will be two or more paths on the super trellis with the same code sequence. In such circumstance, the decoding algorithm has to choose a path randomly. So, it is always possible that there are incorrect estimations of the state path without considering the noise on the channel.

Though MLSD approach can provide the best detection results, its computational complexity is not polynomial (NP-complexity), which means that the computational complexity of MLSD will increase with the user number exponentially. The reason is that the super trellis aligns the users' input and memory together. Thus, the possible size of input alphabet and state number will increase exponentially when the user number increases. As a result, iterative detections are of interests, which have polynomial computation complexity and can approach the performance of MLSD.

2.4.3 Iterative Detection of CFSK

Unlike the optimal detection, which operates on the super trellis and has a single decoder, the iterative detection approach decodes each user's information by their component decoders. After each iteration, the users exchange

their updated information with each others. The obtained information from others are applied in the following iteration. After enough iterations, the decoding results for all the users are supposed to approach the optimal estimation. Two known iterative detection approaches are: interference cancellation and consensus decoder, which treat the updated information differently.

2.4.4 Interference Cancellation (IC)

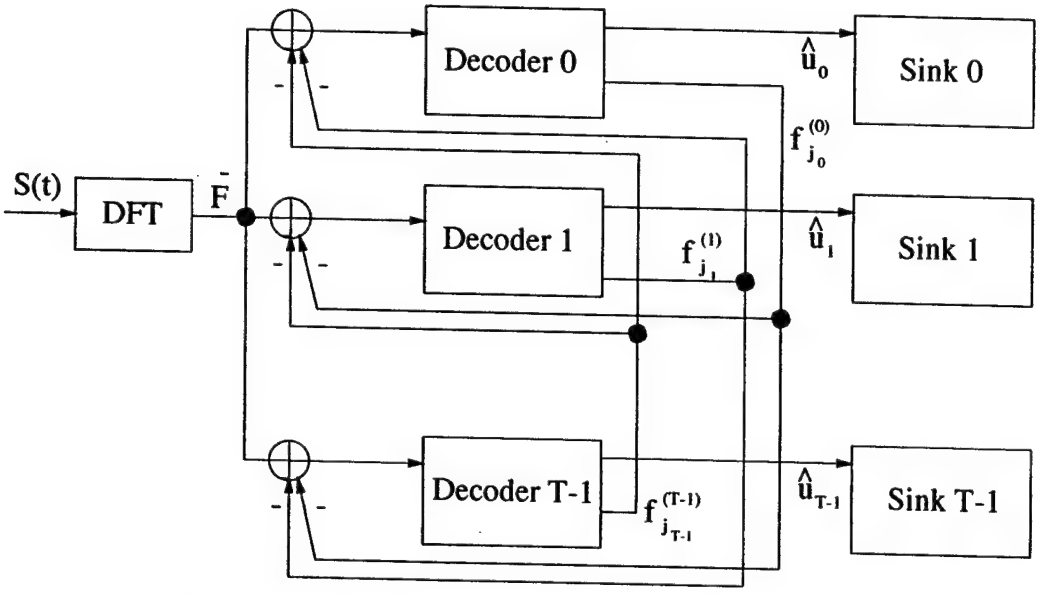

Figure 2.7: The interference cancellation decoder.

The structure of interference cancellation is given in Figure 2.7. In each iteration, the component decoders update the estimation of the transmitted frequencies. At the input of the component decoders, the estimations from

other users are subtracted from the vector $\bar{F}$, so that the input of the component decoder is an updated channel signal for a specific user. Information form other users is canceled from the channel signal $\bar{F}$ as interference. And, each component decoder treats the channel as a single user channel by using single user channel metric. This make it seems that IC is a kind of single user detection. However, for each user, the single user channel is obtained by using the updated estimations of the frequencies from other users. In this sense, IC is regarded as an iterative multiuser detection.

In [23], performance of IC for TCMA system is given, which shows that the performance for IC is much worse than the optimal performance. This is because that the component decoders treat the channel as single user channel and use single user channel metric. So, this project will not put much energy on the IC approach.

2.4.5 The Consensus Decoder

The structure of consensus decoder is given by [18]. In Figure 2.8, the basic structure of consensus decoder is described. The consensus decoder is composed of the users' component decoders. The component decoder takes the DFT output $\bar{F}$ as input and decode each user's information separately. Each component decoder outputs the estimations for each user's symbols and

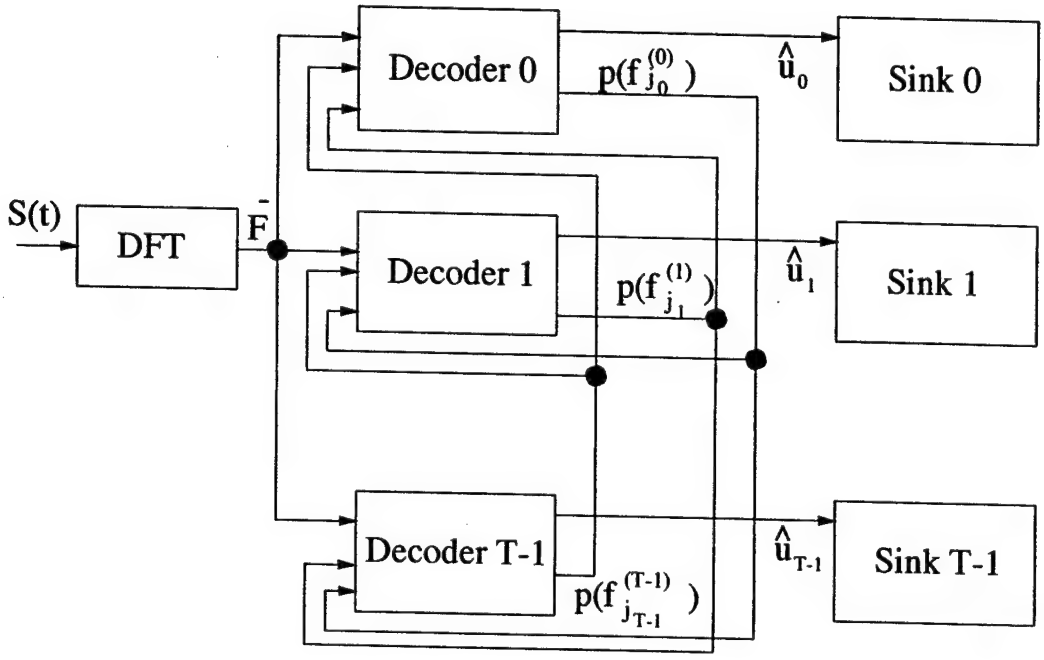

Figure 2.8: The consensus decoder.

the *a posteriori* information on the symbols. For instance, $P(f_{j_u}^{(u)})$ is the *a posteriori* probability for frequency f_{j_u} output by user U_u. The *a posteriori* information is feed to other users' component decoder as *a priori* information. The *a priori* information from other users are applied to update the channel metric during each user's decoding procedure. After enough iterations, the estimation results for each user is supposed to approach the optimal results. In Chapter 4 the consensus decoders using MAP decoder and Turbo iterative decoder are introduced. The performance for the consensus decoders is discussed in Chapter 5.

Chapter 3

Theory and Performance of the CFSK Channel

3.1 Introduction

The collision frequency shift-keyed (CFSK) channel is based on the channel model described by Chang and Wolf as the T-user M-frequency multiple-access channel with and without intensity information [17]. They gave out the capacities of the noiseless channels with and without intensity information. A formal description of the CFSK channel was given by Grant and Schlegel in [21] and [18]. Starting from results in [17], Grant derived the approximated capacity for the noiseless CFSK channel without intensity information in [21], which approaches Chang and Wolf's result asymptotically. Furthermore, Grant provided the upper and lower bounds of the CFSK channel without intensity information in the presence of noise and undetected in-

terference. These results set up a profound theoretical basis for the design of a multiple-access communication system based on the CFSK channel. Due to the complexity of the CFSK channel with intensity information, Grant did not give out the capacity of the noisy CFSK channel with intensity information. Moreover, all the former capacity results are assuming that the users that access to the channel send their symbols synchronously, i.e. the symbols on the channel are aligned with each other perfectly. In practical applications, the synchronization is hard to be achieved among multiple users. For an asynchronous CFSK system, users send their symbols without any cooperation, which means that the phase of the FSK signal sent by a user on a particular frequency is different from those sent by other users. This is similar to introducing a noise to the synchronous channel. Thus, different capacity results of the CFSK channel are anticipated.

In [22] and [31], mathematical models for the synchronous and asynchronous CFSK channels are set up. These models provide a new approach to research the asynchronous CFSK channel. On the other hand, the development of computer technology make it possible to study the more complicated channels, such as the CFSK channel with intensity information. In this work, efforts are made to study the sum capacities for the CFSK channels with and without intensity information in noisy or asynchronous scenarios. Inspired

by ideas in [21], analytical models are set up so that the bounds of the sum capacities are obtained. The upper and lower bounds of the sum capacity of the noisy synchronous CFSK channel with intensity information are computed in section 3, which provides a computer approach for computing the sum capacity. Then, the upper bounds of the noiseless asynchronous CFSK channel with intensity information is obtained by the computer approach. By the analytical methods, an upper bound for the noiseless asynchronous CFSK channel without intensity information is derived. Results show that all the bounds are tight enough for large user numbers. These capacity results are compared together, which provides significant directions in designing a CFSK system for practical applications.

3.2 The CFSK Channel Distribution

3.2.1 Distribution over the Synchronous Channel

According to the channel model in [22] and [31], the output signal of the noisy synchronous CFSK channel on frequency j can be described by

$$\begin{aligned} x_j &= \mu_j\sqrt{E}cos(\phi_j) + n_x, \\ y_j &= \mu_j\sqrt{E}sin(\phi_j) + n_y, \\ \rho_j &= x_j^2 + y_j^2, \end{aligned} \tag{3.1}$$

where μ_j is the number of users that transmit frequency j, E is the symbol energy, ϕ_j is the phase with which all the users transmit the signal, n_x and n_y

are the the Gaussian noise with double-sided power spectral density (P.S.D.) $\frac{N_o}{2}$. Thus, ρ_j is the detected energy on the output of the channel at frequency j. The distribution of ρ_j is given by a non-central chi-square distribution, which is further approximated by

$$f_{\rho_j}(\rho_j) = \frac{1}{N_o} exp\left(- \frac{(\mu_j\sqrt{E} - \sqrt{\rho_j})^2}{N_o} \right). \tag{3.2}$$

The relationship between the above $f_{\rho_j}(\rho_j)$ and $\sqrt{\rho}$ is described in Figure 3.1 for $\mu_j = \{0, 1, \ldots, 8\}$ when the signal-to-noise ratio (SNR) is $8dB$.

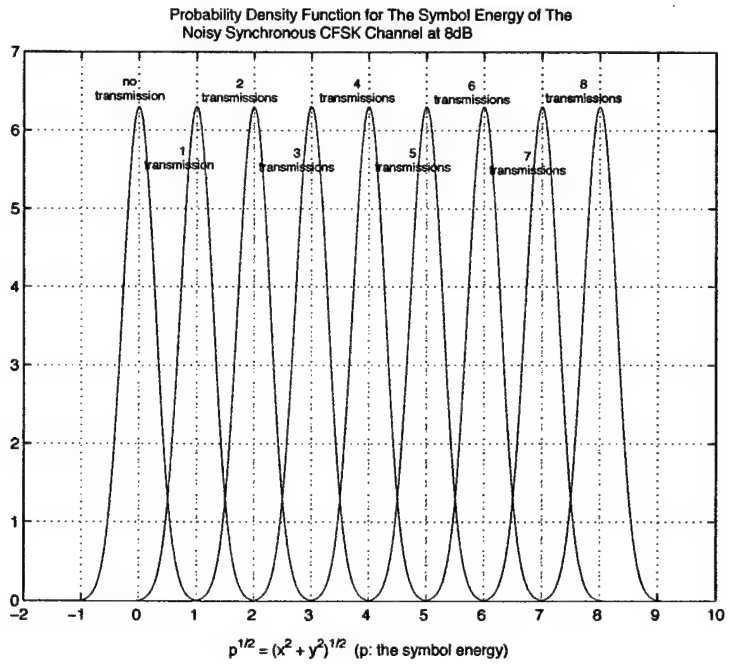

Figure 3.1: P.S.D. of symbol energy of the noisy synchronous CFSK channel.

3.2.2 Distribution over the Asynchronous Channel

For the asynchronous CFSK channel without considering the noise, the output signal given by [22] and [31] is

$$\begin{array}{rcl} x_j & = & \sqrt{E}\sum_{i=1}^{\mu_j} cos(\phi_{j,i}), \\ y_j & = & \sqrt{E}\sum_{i=1}^{\mu_j} sin(\phi_{j,i}), \\ \rho_j & = & x_j^2 + y_j^2, \end{array} \tag{3.3}$$

where $\phi_{j,i}$ denotes the phase with which the user i transmits frequency j. Since all users transmit the signals without any cooperation, the distribution of the phase $\phi_{j,i}$ can be assumed uniform between 0 and 2π as

$$f_\Phi(\phi_{j,i}) = \begin{cases} \frac{1}{2\pi} & 0 \leq \phi_{j,i} \leq 2\pi \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

Theorem 1(Variance of noiseless asynchronous signals): The variance of x_j and y_j given in (3.3) is $\frac{\mu_j E}{2}$, and the signal can be approximated by a noisy synchronous signal as

$$\begin{array}{rcl} x_j & = & \mu_j\sqrt{E}cos(\phi_j) + n_{eff,x}, \\ y_j & = & \mu_j\sqrt{E}sin(\phi_j) + n_{eff,y}, \\ \rho_j & = & x_j^2 + y_j^2, \end{array} \tag{3.5}$$

where the $n_{eff,x}$ and $n_{eff,y}$ are the effective Gaussian noise with double-sided P.S.D. $\frac{\mu_j E}{2}$.

The proof of Theorem 1 is given in Appendix I. From Theorem 1, the distribution of ρ_j over the noiseless asynchronous CFSK channel can be obtained

in a similar way as the one over the noisy synchronous CFSK channel. Thus,

$$f_{\rho_j}(\rho_j) = \frac{1}{\mu_j E} exp\left(- \frac{(\sqrt{\mu_j E} - \sqrt{\rho_j})^2}{\mu_j E} \right). \tag{3.6}$$

The relationship between $f_{\rho_j}(\rho_j)$ and $\sqrt{\rho}$ is described in Figure 3.2 for $\mu_j = \{2, \ldots, 8\}$.

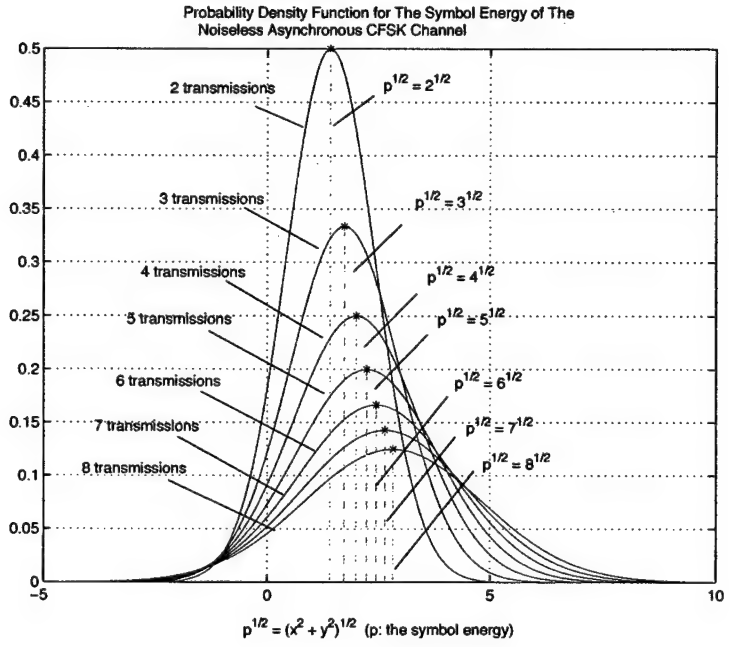

Figure 3.2: P.S.D. of symbol energy of the noiseless asynchronous CFSK channel.

In both Figure 3.1 and Figure 3.2, it can be observed that, the curves of the distributions extend to the area where $\sqrt{\rho}$ is less than 0. This is the result of the approximation of the non-central chi-square distribution, which will not affect later discussion since all integrations are from $-\infty$ to positive values of

$\sqrt{\rho}$. The distribution when μ_j is 1 is not given in Figure 3.2, for the reason that there is no collision on frequency j when $\mu_j = 1$, and distribution of the asynchronous channel is the same as that of the synchronous channel. It can also be observed that, for large μ_j in Figure 3.2, the distributions overlap with each other seriously. This implies that it is much less precise to estimate energy on the asynchronous channel. However, it would be better to detect whether there is transmission on the frequency or not. So, it is reasonable to anticipate that the performance of the asynchronous CFSK channel with intensity information may be worse than that of the asynchronous CFSK channel without intensity information when the system is over-loaded. Later results will prove this anticipation.

3.3 Performance of the Synchronous CFSK Channel

[17] provided the sum capacities of the noiseless synchronous CFSK channel with and without intensity information. These results can be observed in Figure 1.1 when they are compared with the sum capacities of the CDMA systems [18]. Grant provided a tight approximation of the capacity for the noiseless synchronous CFSK channel without intensity information in [21]. Furthermore, upper and lower bounds for the sum capacity of the noisy

synchronous CFSK channel without intensity information is also given in [21]. In this context, discussion on the bounds for the noisy synchronous CFSK channel with intensity information is provided.

Theoretically, the size of the output alphabet from the noisy CFSK channel with intensity information is infinite. This makes it extremely difficult to compute the exact capacity of the noisy CFSK channel with intensity information. Grant's analytical channel model in [21] provides an effective way for computing the upper and lower bounds of the noisy CFSK channel. The idea of the model is to decompose the noisy channel into several concatenated sub-channels. Each sub-channel generates specific errors that would appear on the original channel. The model is modified in this context for computing the upper and lower bounds of the noisy synchronous channel with intensity information.

An analytical model for the noisy synchronous channel with intensity information is given in Figure 3.3. X, Y, U, V and W are the input and output alphabets of the sub-channels. Each element of the alphabets is a $1 \times M$ vector in which each entry denotes the number of transmissions (occupations) on the corresponding frequency. For all $x \in X$, let x be

$$x = (x_1, x_2, \ldots, x_M). \tag{3.7}$$

And, for all $u \in U$, let u be

$$u = (u_1, u_2, \ldots, u_M). \tag{3.8}$$

Then, the single-hop denotes the kind of error that, when the x_j users transmit on frequency j, the output of the single-hop channel on frequency j satisfies that

$$u_j = \begin{cases} x_j & \text{no error happens} \\ x_j \pm 1 & \text{otherwise} \end{cases} \tag{3.9}$$

The alphabet V, which is the output of the noisy channel, can include other errors defined by n-hop for $n > 1$. Let $v \in V$ be defined by

$$v = (v_1, v_2, \ldots, v_M). \tag{3.10}$$

Then, the n-hop error happens when

$$v_j = x_j \pm n. \tag{3.11}$$

In Figure 3.3, the "symmetrifying channel" takes the noised signal as input. At the output of this channel, all kinds of errors will have the same probability p_{sym}.

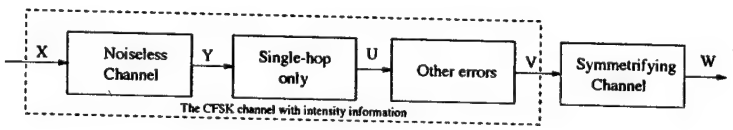

Figure 3.3: The analytical channel model for the CFSK channel with intensity information.

The probability of single-hop, denoted by q , can be computed by the distribution given by (3.2). Then, let the probability p_{sym} be equal to q. According to the data processing inequality [32], the sum capacity of the noisy channel has the following upper and lower bounds:

$$I(W|X) \leq C_{noisy,sum} = I(V|X) \leq I(U|X). \tag{3.12}$$

A lower bound can be computed from $I(W|X)$ as given in the following Theorem. A proof of the similar problem can be found in [21].

Theorem 2: a lower bound of the sum capacity of the noisy synchronous CFSK channel with intensity information is given by

$$C_{noisy,sum} = I(V|X) \geq I(W|X) \geq H(Y) - Mh(q) \geq C_{sum} - Mh(q), \tag{3.13}$$

where C_{sum} is the sum capacity of the noiseless synchronous CFSK channel with intensity information, M denotes the number of frequencies used in the system, $h(q)$ is the binary entropy function given by

$$h(q) = -qlog(q) - (1-q)log(1-q). \tag{3.14}$$

The value $I(U|X)$ provides an upper bound for the sum channel capacity, which can be computed by

$$I(U|X) = H(U) - H(U|X). \tag{3.15}$$

The entropy $H(U)$ is given by

$$H(U) = \sum_{u \in U} -p_U(u) log(p_U(u)). \tag{3.16}$$

The mutual information $H(U|X)$ is given by

$$H(U|X) = \sum_{u \in U, x \in X} p_{U,X}(u,x) I(u|x) = \sum_{u \in U, x \in X} -p_X(x) p_{U|X}(u|x) log(p_{U|X}(u|x)). \tag{3.17}$$

Considering that

$$p_U(u) = \sum_{x \in X} p_X(x) p_{U|X}(u|x), \tag{3.18}$$

it is important to compute the channel transition probability $p_{U|X}(u|x)$. Since U is the output of the sub-channel with single-hop error, $p_{U|X}(u|x)$ is equal to 0 when an output u cannot be generated by a given input x. Thus, for a certain output u of the sub-channel, a set can be constructed as

$$S = \{x | x \in X, \text{u is the output of the sub-channel by input x.}\}. \tag{3.19}$$

As a result, (3.18) becomes

$$p_U(u) = \sum_{x \in S} p_X(x) p_{U|X}(u|x). \tag{3.20}$$

For all $x \in S$, the probability $p_{U|X}(u|x)$ is given by

$$p_{U|X}(u|x) = q^{W(u,x)} (1-q)^{M - W(u,x)}, \tag{3.21}$$

where the function $W(u,x)$ denotes the number of positions in u that are different from those in x. Assume that all users send the frequencies with

equal probability, which is $\frac{1}{M}$. Then, the probability of input symbol x is computed as

$$\begin{aligned} p_X(x) &= \binom{T}{\mu_1}\binom{T-\mu_1}{\mu_2}\cdots\binom{T-\mu_1-\ldots-\mu_{M-1}}{\mu_M}\frac{1}{M^T} \\ &= \frac{T!}{\mu_1!\mu_2!\ldots\mu_M!}\frac{1}{M^T}. \end{aligned} \tag{3.22}$$

By computer approach, it is not difficult to construct the set of all input symbols X. By the rule of single-hop, the set U can also be constructed. When the frequency number, M, and the user number, T, are not very large, the time spent on these constructions is not huge. Following the computations given by (3.15)-(3.22), the upper bound can be obtained. In Figure 3.4, the upper bounds and lower bounds for the system when SNR is $8dB$ are given. It can be found that the bounds are tighter when the number of users gets large. When SNR increase, the bounds will also become tighter. This can be observed in Figure 3.5, where the bounds are given when SNR is $12dB$. The sum capacities of the noiseless channels [17] is also given in Figure 3.4 and 3.5. For high SNR's, those bounds approach the sum capacity of the noiseless channel.

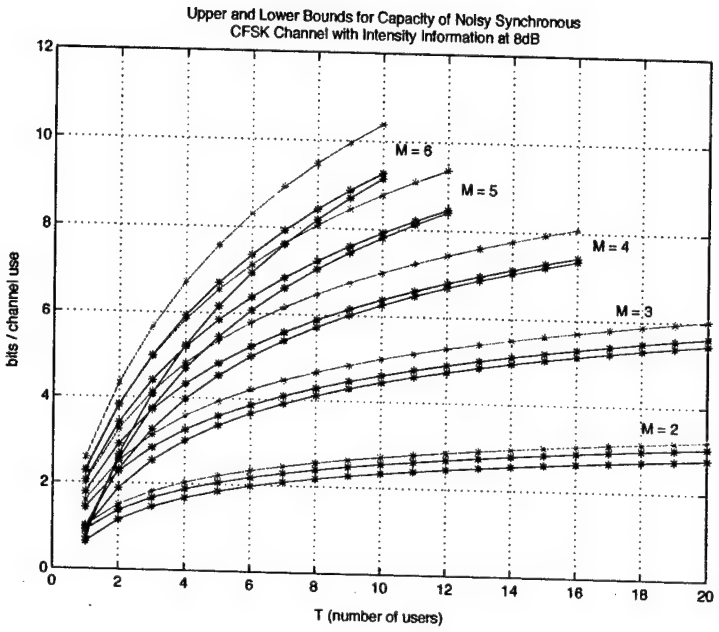

Figure 3.4: Upper and lower bounds for capacity of noisy synchronous CFSK channel with intensity information at 8dB.

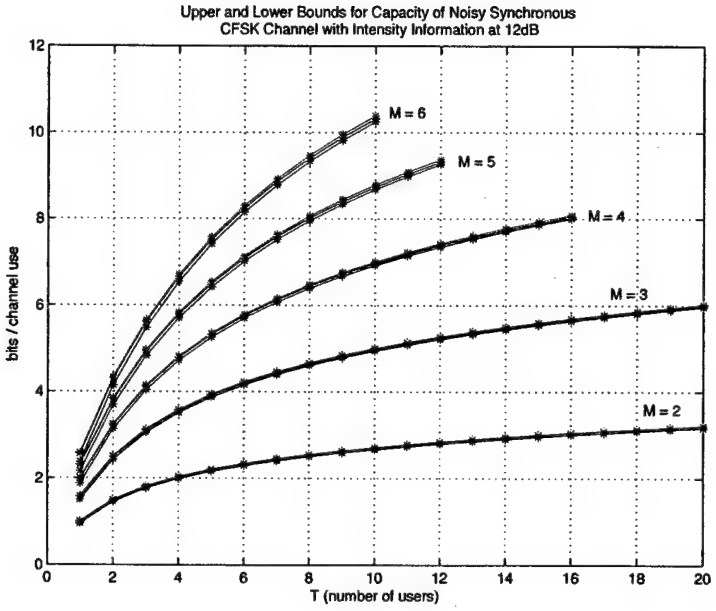

Figure 3.5: Upper and lower bounds for capacity of noisy synchronous CFSK channel with intensity information at 12dB.

3.4 Performance of the Asynchronous CFSK Channel

3.4.1 The Noiseless Asynchronous Channel with Intensity Information

To analyze the capacity of the noiseless asynchronous CFSK channel, the analytical model given in Figure 3.3 is still applied. The basic idea of computing the upper bound of the capacity follows the one given in the previous section. The only difference exists in the probability $p_{U|X}(u|x)$, since the distribution of the channel is given by (3.6). From (3.6), a transmission probability matrix can be constructed as

$$\mathbf{P} = \begin{pmatrix} p(0|0) & p(1,0) & \dots & p(T+\beta|0) \\ p(0|1) & p(1,1) & \dots & p(T+\beta|1) \\ \dots & \dots & \dots & \dots \\ p(0|T) & p(1,T) & \dots & p(T+\beta|T) \end{pmatrix}, \tag{3.23}$$

where $p(u_j|x_j)$ denotes that probability that u_j transmissions are detected at the output of the single-hop channel given that x_j users transmit on frequency j. β is a constant, which is large enough that $p(T+\beta|x_j)$ can be ignored for all $0 \leq x_j \leq T$. Then, the probability $p_{U|X}(u|x)$ is given by

$$p_{U|X}(u|x) = \prod_{j=1}^{M} p(u_j|x_j). \tag{3.24}$$

By computer approach, the alphabets X and U are constructed, and computations given by (3.15)-(3.20), (3.24) and (3.22) are followed. The upper

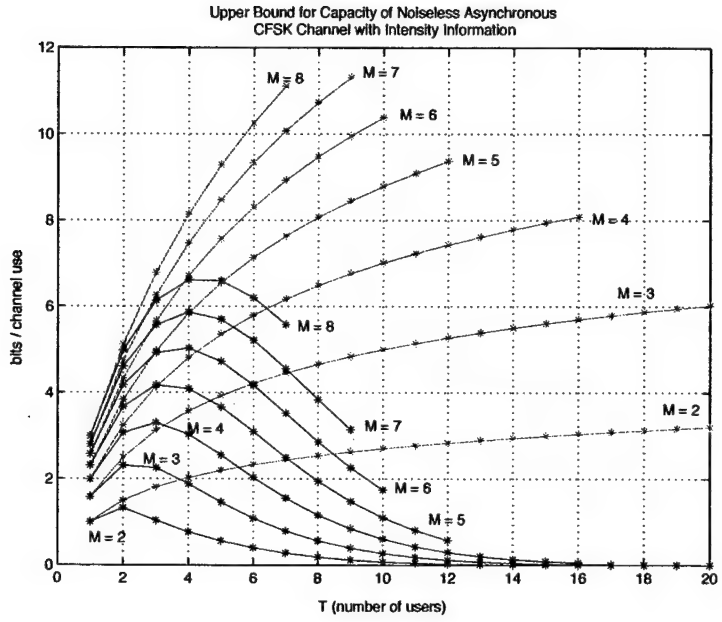

Figure 3.6: Upper bounds for capacity of noiseless asynchronous CFSK channel with intensity information.

bounds when $2 \leq M \leq 8$ are described in Figure 3.6. It can be observed that when user number, T, becomes large, especially when $T \geq M$, the upper bound of the capacity decreases drastically. This is compared to the capacity of the noisy synchronous channel, which keeps increasing when user number increases. And, the upper bound tends to 0 when user number is extremely large. So, the asynchronous channel is unusable under heavy-loaded conditions.

Since the upper bound described in Figure 3.6 contains enough information to show the performance of the channel, and it is too complicated to compute the lower bound from the distribution given by (3.6), the lower bound of the noiseless asynchronous channel with intensity information is not given in this context.

3.4.2 The Noiseless Asynchronous Channel without Intensity Information

To compute the capacity of the noiseless asynchronous CFSK channel without intensity information, an analytical model is given in Figure 3.7. By this model, only upper bound of the capacity is available, which is given by $I(U|X)$. On the first sub-channel, errors only happen when there are 2 transmissions on frequency j ($x_j = 2$). Since the channel is without intensity information, the errors are always 2-hop, i.e. a jump from double-occupation

to inactive.

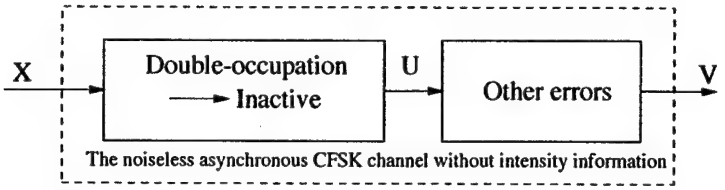

Figure 3.7: The analytical channel model for the noiseless asynchronous CFSK channel without intensity information.

Before computing the upper bound of the channel, some variables are defined as

m: random variable counts active frequencies at the output U;

i: random variable counts active frequencies at the input X;

s: random variable counts single occupied frequencies at the input X;

d: random variable counts double occupied frequencies at the input X.

An upper bound is given by the following theorem, which is proved in Appendix II.

Theorem 3: The upper bound of the noiseless asynchronous channel without intensity information is given by

$$I(U|X) = \sum_{m=0}^{N} p_M(m) log\left(\frac{\binom{N}{m}}{p_M(m)}\right) + h(q) \sum_{d=0}^{\frac{T}{2}} d \sum_{i=1}^{T} p_I(i) p_{D|i}(d|i), \quad (3.25)$$

where

$$p_M(m) = \sum_{i=m}^{T} \sum_{d=1}^{\frac{T}{2}} p_I(i) p_{D|I}(s|i) p_{M|D,I}(m|d,i), \quad (3.26)$$

$$p_{D|I}(d|i) = \frac{\sum_{s=0}^{i} \binom{i}{s}\binom{T}{s} s! \left[\binom{i-s}{d} \prod_{l=0}^{d-1} \binom{T-2l-s}{2} a^{(3)}_{i-s-d,T-s-2d} \right]}{a^{(1)}_{i,T}}, \tag{3.27}$$

$$p_{M|D,I}(m|d,i) = \binom{d}{i-m} q^{i-m}(1-q)^{d-(i-m)}. \tag{3.28}$$

q is the probability of double-occupation to inactive error, which is equal to $p(0|2)$ in (3.23). $a^{(t)}_{n,k}$ denotes that number of ways that put k balls into n bins such that every bin contains at least t balls. $a^{(t)}_{n,k}$ is computed in an iterative way as

$$a^{(1)}_{1,k} = a^{(2)}_{1,k} = a^{(3)}_{1,k} = 1, \tag{3.29}$$

$$a^{(1)}_{n,k} = \sum_{i=0}^{n} (-1)^i \binom{n}{i} (n-i)^k, \tag{3.30}$$

$$a^{(2)}_{n,k} = a^{(1)}_{n,k} - \sum_{j=1}^{n-1} \binom{n}{j}\binom{k}{j} j! a^{(2)}_{n-j,k-j}, \tag{3.31}$$

$$a^{(3)}_{n,k} = a^{(2)}_{n,k} - \sum_{j=1}^{n-1} \binom{n}{j} \prod_{l=0}^{j-1} \binom{k-2l}{2} a^{(3)}_{n-j,k-2j}. \tag{3.32}$$

The upper bounds are described in Figure 3.8 for $2 \geq M \geq 16$. By comparing with the capacity of the noiseless synchronous CFSK channel without intensity information, it is found that the capacity of the asynchronous channel decreases when the number of users gets large, especially when $T \geq M$. For small T, the capacity of the asynchronous channel decreases a little. By results in Figure 3.6 and 3.7, it is found that the upper bounds of the channel without intensity information is generally higher than the bounds of the

channel with intensity information. This is reasonable, since it is less precise to estimate the energy level of the frequency than to detect the active or inactive of the frequency when the users are not synchronized.

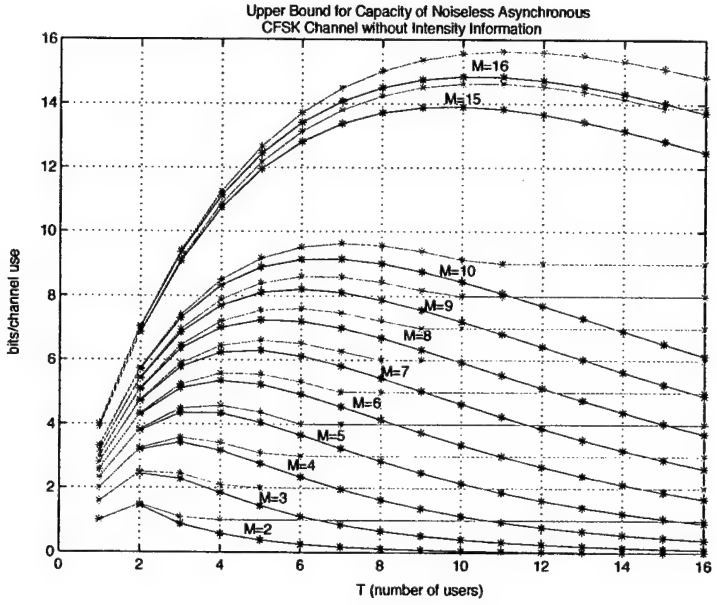

Figure 3.8: Upper bounds for capacity of noiseless asynchronous CFSK channel without intensity information.

3.5 Summary

From the upper bounds given in previous sections, it can be concluded that the asynchronous CFSK channels have poor performance under heavy-loaded conditions. For the channel with intensity information, the capacity approaches 0 when user number is much larger than the frequency number. It

is even worse than the channel without intensity information, since the distributions of signals with large occupation numbers overlap with each other seriously. In this case, it is better to detect the active or inactive of each frequency. When user number is smaller than the frequency number, the synchronous channels and the asynchronous channels behave similarly. Application of the asynchronous channels when the spreading factor (given by $\frac{M}{T}$) is between 1 and 10 is much more significant, since the channels are more efficient under such conditions.

3.6 Appendix I: Proof of Theorem 1

Consider the variable y which is a function of the phase Φ,

$$y = \sqrt{E}sin(\phi), \tag{3.33}$$

where the probability density function (P.D.F.) of the phase Φ is given by $f_\Phi(\phi)$ in (3.4). When $y \leq -\sqrt{E}$, the probability function of y is

$$F_Y(y) = 0. \tag{3.34}$$

When $-\sqrt{E} < y \leq 0$, the probability function of y is

$$\begin{aligned} F_Y(y) &= \int_{\pi-arcsin(y/\sqrt{E})}^{2\pi-arcsin(y/\sqrt{E})} f_\Phi(\phi)d\phi \\ &= \frac{\pi+2arcsin(y/\sqrt{E})}{2\pi}. \end{aligned} \tag{3.35}$$

When $0 < y \leq \sqrt{E}$,

$$\begin{aligned} F_Y(y) &= \int_0^{arcsin(y/\sqrt{E})} f_\Phi(\phi)d\phi + \int_{-arcsin(y/\sqrt{E})+\pi}^{\pi} f_\Phi(\phi)d\phi + \int_\pi^{2\pi} f_\Phi(\phi)d\phi \\ &= \frac{arcsin(y/\sqrt{E})}{2\pi} + \frac{arcsin(y/\sqrt{E})}{2\pi} + \frac{1}{2} \\ &= \frac{\pi+2arcsin(y/\sqrt{E})}{2\pi}. \end{aligned} \tag{3.36}$$

When $y > \sqrt{E}$,

$$F_Y(y) = 1. \tag{3.37}$$

To sum up the above results, the probability function of y is

$$f_Y(y) = y = \begin{cases} 0 & y \leq -\sqrt{E} \\ \frac{\pi+2arcsin(y/\sqrt{E})}{2\pi} & -\sqrt{E} < y \leq \sqrt{E} \\ 1 & y > \sqrt{E} \end{cases} \tag{3.38}$$

Thus, the P.D.F. of Y from $F_Y(y)$ is computed as

$$\begin{aligned} f_Y(y) &= \frac{dF_Y(y)}{dy} \\ &= \begin{cases} \frac{1}{\pi\sqrt{E}} \frac{1}{\sqrt{1-y^2/E}} & -\sqrt{E} < y \leq \sqrt{E} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{3.39}$$

Now, introduce the moment generation function of Y as

$$\begin{aligned} \Phi_Y(s) &= E[e^{sy}] \\ &= \int_{-\sqrt{E}}^{\sqrt{E}} e^{sy} f_Y(y)dy \\ &= \int_{-\sqrt{E}}^{\sqrt{E}} \frac{e^{sy}}{\pi\sqrt{E}\sqrt{1-y^2/E}} dy. \end{aligned} \tag{3.40}$$

Let the variable Ψ be the sum of n variables with P.D.F. $f_Y(y)$. Then, we have

$$\Psi = \sum_{i=1}^{n} y_i = \sum_{i=1}^{n} \sqrt{E} sin(\phi_i), \tag{3.41}$$

where the P.D.F. of ϕ_i is $f_\Phi(\phi)$ in (3.4). So, the moment generation function of Ψ can be obtained from $\Phi_Y(s)$ as

$$\Phi_\Psi(s) = [\Phi_y(s)]^n = \left[\int_{-\sqrt{E}}^{\sqrt{E}} \frac{e^{sy}}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy\right]^n. \tag{3.42}$$

From the moment generation function $\Phi_\Psi(s)$, the expectation and variance of Ψ can by obtained. Firstly, the expectation of Ψ is

$$\begin{aligned} E[\psi] &= \left.\frac{d\Phi_\Psi(s)}{ds}\right|_{s=0} \\ &= n\left[\int_{-\sqrt{E}}^{\sqrt{E}} \frac{e^{sy}}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy\right]^{n-1} \left.\int_{-\sqrt{E}}^{\sqrt{E}} \frac{ye^{sy}}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy\right|_{s=0} \\ &= n\left[\int_{-\sqrt{E}}^{\sqrt{E}} \frac{1}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy\right]^{n-1} \int_{-\sqrt{E}}^{\sqrt{E}} \frac{y}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy \\ &= 0. \end{aligned} \tag{3.43}$$

Then, the variance of Ψ is

$$\begin{aligned} var[\Psi] &= E[(\Psi - E[\Psi])^2] \\ &= E[\Psi^2] \\ &= \left.\frac{\partial^2\Phi_\Psi(s)}{\partial s^2}\right|_{s=0} \\ &= n(n-1)\left[\int_{-\sqrt{E}}^{\sqrt{E}} \frac{e^{sy}}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy\right]^{n-2}\left[\int_{-\sqrt{E}}^{\sqrt{E}} \frac{ye^{sy}}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy\right]^2 \\ &\quad +n\left[\int_{-\sqrt{E}}^{\sqrt{E}} \frac{e^{sy}}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy\right]^{n-1} \left.\int_{-\sqrt{E}}^{\sqrt{E}} \frac{y^2e^{sy}}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy\right|_{s=0} \\ &= n\left[\int_{-\sqrt{E}}^{\sqrt{E}} \frac{1}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy\right]^{n-1} \int_{-\sqrt{E}}^{\sqrt{E}} \frac{y^2}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy \\ &= n\left[\frac{1}{\pi}\left[arcsin(1) - arcsin(-1)\right]\right]^{n-1} \int_{-\sqrt{E}}^{\sqrt{E}} \frac{y^2}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy \\ &= n\int_{-\sqrt{E}}^{\sqrt{E}} \frac{y^2}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy. \end{aligned} \tag{3.44}$$

Consider the last term of the above equation,

$$\begin{aligned}
\int_{-\sqrt{E}}^{\sqrt{E}} \frac{y^2}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy &= 2E\int_0^{\sqrt{E}} \frac{1+y^2/E-1}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy \\
&= 2E\int_0^{\sqrt{E}} \frac{1}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy - \frac{2\sqrt{E}}{\pi}\int_0^{\sqrt{E}}\sqrt{1-y^2/E}dy \\
&= \frac{2E}{\pi}arcsin(y/\sqrt{E})\Big|_0^{\sqrt{E}} - \frac{2\sqrt{E}}{\pi}\int_0^{\sqrt{E}}\sqrt{1-y^2/E}dy \\
&= E - \frac{2\sqrt{E}}{\pi}y\sqrt{1-y^2/E}\Big|_0^{\sqrt{E}} + \frac{2\sqrt{E}}{\pi}\int_0^{\sqrt{E}} y\frac{-2y/E}{2\sqrt{1-y^2/E}}dy \\
&= E - 2\int_0^{\sqrt{E}} \frac{y^2}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy \\
&= E - \int_{-\sqrt{E}}^{\sqrt{E}} \frac{y^2}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy.
\end{aligned} \tag{3.45}$$

So,

$$\int_{-\sqrt{E}}^{\sqrt{E}} \frac{y^2}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy = \frac{E}{2}. \tag{3.46}$$

At last, the variance of Ψ is

$$var[\Psi] = E[\Psi^2] = n\int_{-\sqrt{E}}^{\sqrt{E}} \frac{y^2}{\pi\sqrt{E}\sqrt{1-y^2/E}}dy = \frac{nE}{2}. \tag{3.47}$$

If the variable y is a cosine function of the phase ϕ, say

$$y = \sqrt{E}cos(\phi), \tag{3.48}$$

and let the variable Ψ be

$$\Psi = \sum_{i=1}^{n} y_i = \sum_{i=1}^{n} \sqrt{E}cos(\phi_i), \tag{3.49}$$

where the P.D.F. of ϕ_i is $f_\Phi(\phi)$, the $E[\Psi]$ and $var[\Psi]$ can be computed in the similarly manner as

$$E[\Psi] = 0, \tag{3.50}$$

$$var[\Psi] = \frac{nE}{2}. \tag{3.51}$$

For the noiseless asynchronous CFSK channel, the signal on frequency j is given by (3.3). Thus, we have

$$\begin{array}{rcl} var[x_j] & = & E[x_j^2] = \frac{\mu_j E}{2}, \\ var[y_j] & = & E[y_j^2] = \frac{\mu_j E}{2}. \end{array} \tag{3.52}$$

So, the expectation of ρ_j is

$$E[\rho_j] = E[x_j^2 + y_j^2] = E[x_j^2] + E[y_j^2] = \mu_j E. \tag{3.53}$$

Assume that there is some phase ϕ_j. And, let the effective Gaussian noise be $n_{eff,x}$ and $n_{eff,y}$ with double-sided P.S.D. $\frac{\mu_j E}{2}$. According to large number theorem, the signal on the noiseless asynchronous channel can be approximated by a signal on the synchronous channel with Gaussian noise when μ_j is large. And, the effective signal is approximated by

$$\begin{array}{rcl} x_j & = & \sqrt{\mu_j E} cos(\phi_j) + n_x, \\ y_j & = & \sqrt{\mu_j E} sin(\phi_j) + n_y. \end{array} \tag{3.54}$$

3.7 Appendix II:Proof of Theorem 3

From Figure 3.7, it is obvious that an upper bound of the sum capacity of the noiseless asynchronous CFSK channel without intensity information is given by $I(U|X)$ where

$$I(U|X) = H(U) - H(U|X). \tag{3.55}$$

Since m is the number of active frequencies at the output U, and the probability of m is $p_M(m)$, then

$$H(U) = \sum_{m=0}^{M} p_M(m) log\left(\frac{\binom{M}{m}}{p_M(m)}\right). \tag{3.56}$$

Proof of (3.56), (3.30) and (3.31) can be found in [21]. And, (3.26) is obvious. To obtain $a_{n,k}^{(3)}$, we need to consider $a_{n,k}^{(2)}$, which denotes the number of ways that put k balls into n bins so that each bin contains at least 2 balls. By subtracting the cases that there are bins that contain exactly 2 balls, $a_{n,k}^{(3)}$ can be obtained. For the n bins and k balls, if there are j bins that contain exact 2 balls, there are $\binom{n}{j}\prod_{l=1}^{j-1}\binom{T-2l}{2}$ ways to put the $2j$ balls in to the j bins. For each of these settlements, there are $a_{n-j,k-2j}^{(3)}$ ways to put the others balls into the remained bins. So, for the $a_{n,k}^{(2)}$ cases, the number of cases that there are j bins contain exactly 2 balls is given by

$$N_{n,k,j}^{(2)} = \binom{n}{j}\prod_{l=1}^{j-1}\binom{T-2l}{2} a_{n-j,k-2j}^{(3)}. \tag{3.57}$$

As a result, $a_{n,k}^{(3)}$ is computed by

$$a_{n,k}^{(3)} = a_{n,k}^{(2)} - \sum_{j=1}^{n-1} N_{n,k,j}^{(2)}. \tag{3.58}$$

By substituting (3.57) in (3.58), (3.32) is obtained.

The probability $p_{D|I}(d|i)$ is equal to the probability that there are d bins that contain exactly 2 balls by putting T balls into i bins without empty. The

number of ways that put T balls into i bins without empty is $a_{i,T}^{(1)}$. For the cases that there are s bins that contain exactly 1 ball, and d bins contain exactly 2 balls, the number of such settlements is

$$N_{i,T,s,d}^{(1,2)} = \binom{i}{s}\binom{T}{s} s! N_{i-s,T-s,d}^{(2)}. \tag{3.59}$$

Then, $p_{D|I}(d|i)$ is given by

$$p_{D|I}(d|i) = \frac{\sum_{s=0}^{i} N_{i,T,s,d}^{(1,2)}}{a_{i,T}^{(1)}}, \tag{3.60}$$

when $d \leq i$. For $d > i$, $p_{D|I}(d|i)$ is 0. By substituting (3.59) into (3.60), (3.27) is obtained.

Consider the model given in Figure 3.7, the errors happen only when the double occupied frequencies jump to inactive. Such error is with the probability $q = p(0|2)$, where $p(0|2)$ is given in (3.23). For the reason that no inactive frequency that can be activated, if there are m active frequencies at the output U, then we must have $m \leq i$ and there are $i - m$ frequencies are deactivated, which are double occupied at input X. So, there are $\binom{d}{i-m}$ possible cases that $i - m$ of d double occupied frequencies are deactivated. For each of them, the probability is $q^{i-m}(1-q)^{d-(i-m)}$. So, $p_{M|D,I}(m|d,i)$ is given by (3.28).

A known fact before computing the value $H(U|X)$ is that the only error at U is caused by the deactivation of the double occupied frequencies. Hence,

each of the d double occupied frequencies is considered as a separated binary channel with transition probability q. Let $E_D(d)$ be the expectation of d, then we have

$$H(U|X) = h(q)E_D(d). \tag{3.61}$$

The expectation $E_D(d)$ is given by

$$E_D(d) = \sum_{d=0}^{\frac{T}{2}} d \sum_{i=1}^{T} P_I(i) P_{D|I}(d|i). \tag{3.62}$$

By substituting (3.62), (3.61) and (3.56) into (3.55), the upper bound (3.25) is obtained.

Chapter 4

Multiuser Detection on the CFSK Channel

4.1 Introduction

For a multiple access system, the multiuser detection operates on the channel output that includes information from all the users. For a particular user, the receiver does not treat other users as unwanted noise. On the other hand, when the receiver is making decision on one of the users, it takes information from other users as useful information.

As introduced in Section 2.4, there are two categories of multiuser detection for the CFSK system. The optimal approach takes all users as a single entity by constructing a super-trellis, and does MLSD on the super-trellis. Due to complexity reasons, the optimal approach is not practical for large

user number applications. Thus, suboptimal approaches are required so that lower decoding complexity can be achieved by sacrificing some system performance. Iterative detection can provide suboptimal performance by iterative decoding structure based on each user's component decoder. Complexity of the iterative detection is decided by complexity of the component decoder. Two known iterative detections are interference cancellation and consensus decoder. Known results show that there is a large distance between the performance of interference cancellation and that of the optimal approach [14]. In this context, the consensus decoder will be discussed in detail.

Basic structure of the consensus decoder has been given in Figure 2.8. Each user has a component decoder in the consensus decoder. The component decoder detects the user's code sequence based on the user's code structure. However, it does not have any knowledge about other user's code structures. This is different from the MLSD approach, where all users' code structures are considered together. Thus, it seems that the consensus decoder has no difference from the single-user detection. In Figure 2.8, each component decoder takes the *a priori* information on the symbols from other users as parameters. These parameters are used in computing the channel metric. In this sense, the consensus decoder does multiuser detection by exchanging the *a priori* information among users and using the users' information in

computing the channel metric.

In this chapter, the optimal multiuser detection approach using MLSD is introduced first. Then, the iterative detection using consensus decoder is discussed. Based on different codes and decoding algorithms, the consensus decoder is classified into the MAP-consensus decoder and the Turbo-consensus decoder. The channel metric in the MAP algorithm is derived, which introduced high complexity to the consensus decoder when the system load is high. Thus, some simplification approaches for the channel metric is discussed in the end.

4.2 Maximal Likelihood Sequence Detection

4.2.1 Channel Metric

The optimal multiuser detection operates on the received signal, which can be described by the vector $\bar{F}$ given in (2.7). The decision is made by choosing a point in the signal space which is the closest in Euclidean distance to the received signal expressed by $\bar{F}$. An example of the signal space is given in Figure 2.3. The points in the signal space can be described by

$$\bar{\mu} = (\mu_0, \mu_1, \ldots, \mu_{M-1}), \tag{4.1}$$

where μ_j describes the number of transmissions on frequency f_j. In Chapter 3, $f_{\rho_j}(\rho_j)$ in (3.2) gives a measurement of the distance on the j^{th} dimension between the received signal and the point in the signal space. If all the M dimensions are considered, then the Euclidean distance can be computed by

$$D(\bar{F}, \bar{\mu}) = \left(\frac{1}{N_o}\right)^M \prod_{j=0}^{M-1} e^{\left(-\frac{(\mu_j\sqrt{E_s}-\sqrt{\rho_j})^2}{N_0}\right)}. \tag{4.2}$$

Thus, the channel metric for MLSD is given by $D(\bar{F}, \bar{\mu})$.

4.2.2 MLSD on the Super-Trellis

The channel metric given in (4.2) implies that the detection algorithm must consider all users' code structures together. This means that a super-trellis based on all users' trellises should be constructed. In [14], method for constructing the super-trellis code is described. The input bits of the super code is given by aligning all users input bits together. State of the users' trellis are also aligned together. For the CFSK system, the output of the super code can be expressed by the vector $\bar{\mu}$ in (2.2.3), which denotes the points in the signal space.

For convenience of discussion, the example of the 2-user super-trellis in Figure 2.4 is described here again in Figure 4.1. Each user uses a rate 1/2, memory 1 code. Thus, the super-code has 2 input bits and 2 memories. For the super trellis, the state number is 4. The branch output is constructed by the

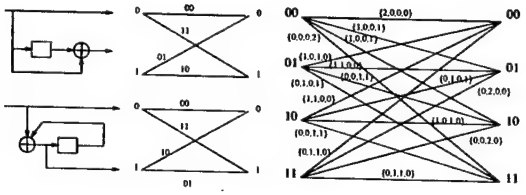

Figure 4.1: A two-user super-trellis.

output of each user's code. Since there are 4 frequencies, the output symbol is a vector of 4 entries. Each entry denotes the number of users that transmit on the frequency. Using the metric given in (4.2), a MLSD procedure can work on the super-trellis and give the optimal detection results. Note that MLSD can be implemented by the Viterbi algorithm.

A limitation of the MLSD approach is that the complexity of the super-trellis is extremely high when there are a large number of users. Though the number of input bits k and memory m of the super code increases linearly with the user number, the input alphabet and possible states increases exponentially with k and m. Thus, complexity of the super-trellis grows exponentially with the number of users. This is reason that the suboptimal detection using iterative algorithm is applied.

4.3 Iterative Detection

4.3.1 Channel Metric for the Consensus Decoder

In Figure 2.8, the component decoder takes the signal $\bar{F}$ as input, and makes decision by choosing a point in the signal space that is the closest in Euclidean distance with $\bar{F}$. A difference from the MLSD is that the component decoder behaves on the single user's code structure. In this case, each user has a single-user signal space, which is different from that described in Figure 2.3. Since all users use the same M-ary FSK modulation, their single-user signal space is an M-dimensional space. The j^{th} point in the space has the coordinate $(0, \ldots, 0, 1, 0, \ldots, 0)$, where the 1 is in the j^{th} position of the coordinate. There is no way to compute directly the distance between a point in the single-user signal space and the received signal that is in the multiuser signal space. However, the probability that a point in the single-user signal space leads to the received signal can be computed. In other words, the channel transition probability $p(\bar{F}|f_{j_u}^{(u)})$ can be computed. This probability can act as a metric of the distance between the point in the single-user signal space and the received signal. Computation of this metric is possible because that the component decoder takes other users' information on the frequencies as parameters.

Denote the set of users by

$$\mathcal{U} = \{U_0, U_1, \ldots, U_{T-1}\}. \tag{4.3}$$

Assume user U_0 chooses frequency $f_{j_0}^{(0)}$. If user U_u chooses a frequency $f_{j_u}^{(u)}$ with probability $p(f_{j_u}^{(u)})$, choices of the other $T-1$ users amount to a frequency combination

$$c_u = (f_{j_0}^{(0)}, \ldots, f_{j_{u-1}}^{(u-1)}, f_{j_{u+1}}^{(u+1)}, \ldots, f_{j_{T-1}}^{(T-1)}). \tag{4.4}$$

Then, the probability for the frequency combination c_0 given that user U_0 transmits on frequency $f_{i_0}^{(0)}$ is

$$p(c_0|f_{j_0}^{(0)}) = \prod_{u=1}^{T-1} p(f_{j_u}^{(u)}). \tag{4.5}$$

Given that user U_0 transmits $f_{j_0}^{(0)}$ and the frequency combination c_0 is transmitted by the other $T-1$ users, the multiplicity factor μ_j for each frequency f_j can be obtained. For the synchronous transmission, the distribution of the received energy ρ_j for f_j is given by (3.2), when $f_{j_0}^{(0)}$ and c_0 are known. Thus, the probability of ρ_j given $f_{j_0}^{(0)}$ and c_0 is

$$p(\rho_j|c_0, f_{j_0}^{(0)}) = \frac{1}{N_0} e^{\left(-\frac{(\mu_j\sqrt{E_s}-\sqrt{\rho_j})^2}{N_o}\right)}. \tag{4.6}$$

If the transmission of each frequency in the set S (see (2.4)) is independent, the probability of receiving frequency vector $\bar{F}$ given $f_{j_0}^{(0)}$ and c_0 is

$$p(\bar{F}|c_0, f_{j_0}^{(0)}) = \left(\frac{1}{N_o}\right)^M \prod_{j=0}^{M-1} e^{\left(-\frac{(\mu_j\sqrt{E_s}-\sqrt{\rho_j})^2}{N_o}\right)}. \tag{4.7}$$

Let all the possible combinations c_0 be in the set C_0. Then, the probability of the received frequency vector $\bar{F}$ given that U_0 transmits $f_{j_0}^{(0)}$ can be represented by

$$p(\bar{F}|f_{j_0}^{(0)}) = \sum_{c_0 \in C_0} P(c_0|f_{i_0}^{(0)}) P(\bar{F}|c_0, f_{j_0}^{(0)}). \qquad (4.8)$$

After substituting the probabilities $P(c_0|f_{j_0}^{(0)})$ and $P(\bar{F}|c_0, f_{j_0}^{(0)})$ in (4.5) and (4.7), the channel transition probability of the CFSK channel with intensity information for the synchronous transmission is

$$p(\bar{F}|f_{j_0}^{(0)}) = \left(\frac{1}{N_o}\right)^M \sum_{i_1=0}^{M-1} \sum_{i_2=0}^{M-1} \cdots \sum_{i_{T-1}=0}^{M-1} \left[\prod_{u=1}^{T-1} p(f_{i_u}^{(u)}) \prod_{j=0}^{M-1} e^{\left(-\frac{(\mu_j \sqrt{E_s} - \sqrt{\rho_j})^2}{N_o}\right)} \right] \qquad (4.9)$$

where i_u is the index of the frequency for the u^{th} user.

In Chapter 2, the distribution of the received energy ρ_j for noisy asynchronous channel is not given. Assume the asynchronous channel is noiseless, then the probability of ρ_j given $f_{j_0}^{(0)}$ and c_0 is

$$p(\rho_j|c_0, f_{j_0}^{(0)}) = \frac{1}{\mu_j E_s} e^{\left(-\frac{(\sqrt{\mu_j E_s} - \sqrt{\rho_j})^2}{\mu_j E_s}\right)}. \qquad (4.10)$$

If the AWGN is introduced to the channel with double-sided P.S.D. N_o, then the over variance of ρ_j can be approximated by $N_o + \mu_j E$. When the user number is smaller than the frequency number, the average value of μ_j is lower than 1. Especially, when the spreading factor is high, average value of μ_j tends to 0. In this case, the probability (4.10) can be approximated by

$$p(\rho_j|c_0, f_{j_0}^{(0)}) = \frac{1}{N_o} e^{\left(-\frac{(\sqrt{\mu_j E_s} - \sqrt{\rho_j})^2}{N_o}\right)}. \qquad (4.11)$$

As a result, the channel transition probability of the CFSK channel with intensity information for the asynchronous transmission is

$$p(\bar{F}|f_{j_0}^{(0)}) = \left(\frac{1}{N_o}\right)^M \sum_{i_1=0}^{M-1}\sum_{i_2=0}^{M-1}\cdots\sum_{i_{T-1}=0}^{M-1}\left[\prod_{u=1}^{T-1} p(f_{j_u}^{(u)}) \prod_{j=0}^{M-1} e^{\left(-\frac{(\sqrt{\mu_j E_s}-\sqrt{\rho_j})^2}{N_o}\right)}\right] \quad (4.12)$$

4.3.2 The MAP Consensus Decoder

In the transmitters of the CFSK system, if the convolutional codes are used for the users, it is possible for the consensus decoder to use the MAP [33] algorithm for the component decoders. An important reason for using the MAP algorithm is that the component decoders need to output the *a posteriori* probabilities for the symbols. As discussed previously, these *a posteriori* probabilities are exchanged to other users' component decoders as the *a priori* probability in computing the channel metric given in (4.9) and (4.12). This consensus decoder is thus called MAP-consensus decoder.

In Figure 4.2, the structure of the MAP-consensus decoder is described. For the first iteration, all users initialize the distribution of the *a priori* probabilities by the uniform distribution. In other words, $p(f_{j_u}^{(u)})$ is initialized to $\frac{1}{M}$ for all users and all frequencies. After the iteration, the *a posterior* probabilities $p(f_{j_u}^{(u)}|\bar{F})$ can be obtained. Instead of making a decision on the output symbols instantly, these *a posterior* probabilities are fed back to other users' decoders, and are used as the *a priori* probabilities in computing the

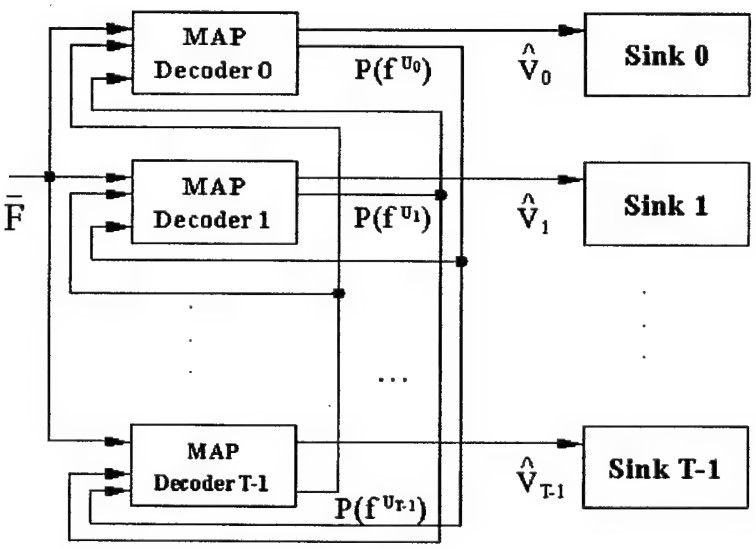

Figure 4.2: The consensus decoder using the MAP algorithm.

metric. So, the channel metric $p(\bar{F}|f_{j_u}^{(u)})$ in the next iteration is updated and improved.

To obtain the *a posteriori* probability $p(f_{j_u}^{(u)}|\bar{F})$, all operations in the MAP algorithm are symbol-based. Thus, the final decision is made on the code symbols. This is the reason that the recursive systematic codes (RSC) are used in the system, since we can obtain the information bits directly from the symbols. Another reason for application of the RSC is that it is easy to avoid the catastrophic codes when we randomly generate the RSC codes. [34] provides a symbol-based MAP algorithm for decoding the RSC codes. Let the length of the code be τ. The received signal from start to the current stage is denoted by $\bar{F}_1^t$, where t denotes the current stage. Denote the current state on the trellis by S_t, which takes the value m. The trellis branches output symbols X_t with the value s, which belongs to the set $\mathcal{S}$ in (2.4). And, let the current received signal be $\bar{F}_t$. The code symbol output by Then, the forward and backward recursions are

$$\begin{aligned} \alpha_t(m) &= p(S_t = m, \bar{F}_1^t) \\ &= \frac{\sum_{m'} \sum_{s\in\mathcal{S}} \gamma_s(\bar{F}_t, m', m)\alpha_{t-1}(m')}{\sum_m \sum_{m'} \sum_{s\in\mathcal{S}} \gamma_s(\bar{F}_t, m', m)\alpha_{t-1}(m')}, \end{aligned} \tag{4.13}$$

$$\begin{aligned} \beta_t(m) &= p(\bar{F}_{t+1}^{\tau}|S_t = m) \\ &= \frac{\sum_{m'} \sum_{s\in\mathcal{S}} \gamma_s(\bar{F}_{t+1}, m, m')\beta_{t+1}(m')}{\sum_m \sum_{m'} \sum_{s\in\mathcal{S}} \gamma_s(\bar{F}_t, m', m)\alpha_t(m')}. \end{aligned} \tag{4.14}$$

The branch transition probabilities are given by

$$\begin{aligned}\gamma_s(\bar{F}_t, m', m) &= p(S_t = m, Y_t = \bar{F}_t, X_t = s | S_{t-1} = m') \\ &= p(Y_t = \bar{F}_t | X_t = s, S_{t-1} = m', S_t = m) q(X_t = s | S_{t-1} = m', S_t = m) \\ &\quad R(S_t = m | S_{t-1} = m').\end{aligned} \tag{4.15}$$

We can also have the probability

$$\begin{aligned}\delta_t(m, m') &= p(S_{t-1} = m', S_t = m, \bar{F}_1^\tau) \\ &= \alpha_t(m') \gamma_s(\bar{F}_t, m', m) \beta_t(m).\end{aligned} \tag{4.16}$$

Thus, the *a posteriori* probability of the symbols are

$$p(X_t = s | Y_1^\tau = \bar{F}_1^\tau) = \frac{p(X_t = s, \bar{F}_1^\tau)}{p(\bar{F}_1^\tau)}. \tag{4.17}$$

By normalization, the value $p(\bar{F}_1^\tau)$ is cancelled out from the above probability.

Thus, the *a posteriori* probabilities can be written as

$$\begin{aligned}p(X_t = s | \bar{F}_1^\tau) &= \Lambda_t(s) \\ &= p(X_t = s, \bar{F}_1^\tau) \\ &= \textstyle\sum_{(m', m) \in B(s)} \delta_t(m, m') \\ &= \textstyle\sum_{(m', m) \in B(s)} \alpha_t(m') \gamma_s(\bar{F}_t, m', m) \beta_t(m).\end{aligned} \tag{4.18}$$

The set $B(s)$ denotes the set of branch transitions from state m' to m with output symbol s. The decision is made by choosing a symbol s with the maximum value for $\Lambda_t(s)$. The *a posteriori* information $p(X_t = s | \bar{F}_1^\tau)$ for all symbols is exchanged to other users as the *a priori* information $p(X_t = s)$ for the symbols.

4.3.3 The Turbo Consensus Decoder

To improve the error correction capability of the CFSK system, Turbo codes can be applied in the system. Thus, both the transmitter and the consensus-decoder need to be modified. Since the consensus decoder does symbol-by-symbol detection based on the received frequency vector, the corresponding turbo encoder should also be symbol based. Robertson introduced a band-width efficient coding scheme employing turbo codes [35], which is adapted by the turbo consensus decoder in this context.

In a CFSK system using the Turbo-consensus decoder, the transmitter should also use the Turbo code [36], which is also called the parallel concatenation convolutional code. The encoder structure of Robertson's scheme has some differences from the Berrou's encoder, as given in Figure 4.3. The interleaver is symbol based. The puncturing is also symbol-based. Thus, the systematic bits generated by the second encoder will be sent out with the codeword if the corresponding symbol is not punctured. For this reason, the interleaver should interleave an even indexed symbol to an even indexed position, and interleave an odd indexed symbol to an odd indexed position, so that the two codewords generated by the two encoders with the same systematic bits will not be both sent out or both punctured. After puncturing, the codewords will be mapped to the corresponding frequencies, which is the same as we

have discussed for the convolutional encoder.

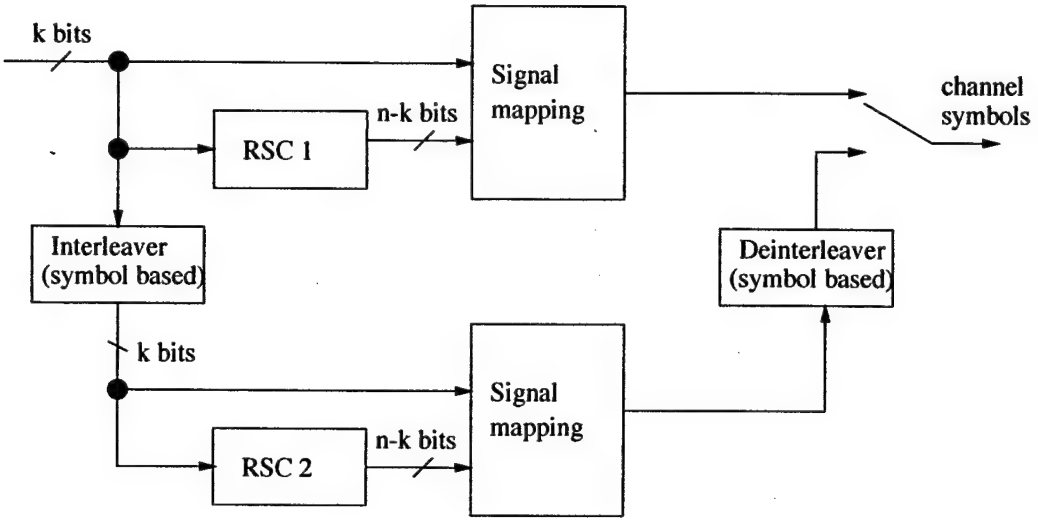

Figure 4.3: Symbol based Turbo encoder

In the Turbo-consensus decoder, each user's Turbo iterative decoder is shown in Figure 4.4. The received signal $\bar{F}$ is fed into the two component decoders alternatively according to the puncturing rules at the encoder. The component decoders are using MAP algorithm. The received signal is used to compute the channel metric. If the corresponding symbol is punctured, the metric is set to 1 since nothing can be obtained from the channel. The MAP algorithm is symbol-based as the one used for the MAP-consensus decoder. Computation of $\alpha_t(m)$, $\beta_t(m)$ and $\gamma_s(\bar{F}_t, m', m)$ is according to (4.13), (4.14) and (4.15). Also, the *a posteriori* probability of a symbol is obtained by (4.18). Both component decoders output $p(X_t = s|\bar{F}_1^T)$. They are exchanged

to other users' decoders for computing the metric. Note that the *a posteriori* probabilities generated by the outer decoder should be exchanged to outer decoder, and the those generated by the inner decoder should be exchanged to the inner decoder. The *a posteriori* probability generated by outer decoder (or inner decoder) will also be feed to the inner decoder (or outer decoder) of the same user for computing the extrinsic information of the corresponding systematic bits (thus to compute $\gamma_t^s(m', m)$). In computing the extrinsic information, the input *a priori* information of this component decoder should be taken from the output *a posteriori* information of this component decoder, so that the information for the systematic bits is not reused in later iterations of the Turbo decoding. At last, the decision can be made on the output symbol by both the inner component decoder or the outer component decoder. In this project, the decision is made by inner component decoder.

4.4 Computational Complexity Problems in the Consensus Decoder

4.4.1 Computational Complexity of the Channel Metric

In computing the channel metric $p(\bar{F}|f_{j_0}^{(0)})$ by (4.8), the summation is over the set C of all possible frequency combinations. Since each of the $T-1$

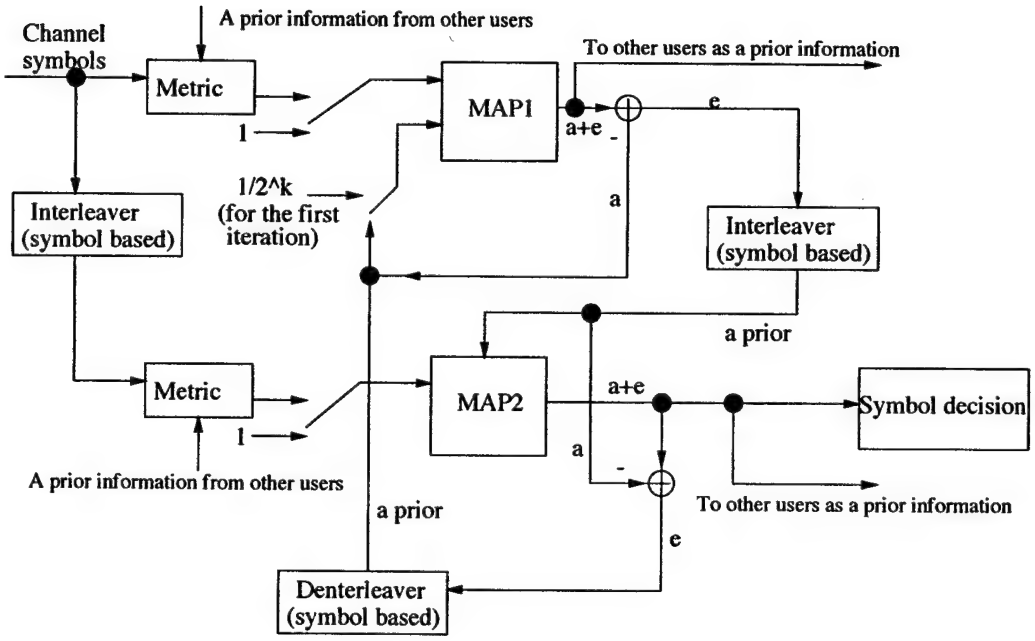

Figure 4.4: Symbol based Turbo iterative decoder

users has M possible choices for picking a frequency from the set S, there are M^{T-1} possible frequency combinations in the set C. Thus, the computational complexity of $p(\bar{F}|f_{j_0}^{(0)})$ increases exponentially with the user number T. The extremely high computational complexity in highly loaded systems is impractical for realization of the consensus decoder. So, approximation of the metric with a reduced complexity is critical for the application of a CFSK system.

Ideally, the computational complexity of the approximated metric should increase linearly with the number of users. By observing the structure of the consensus decoder, it is found that most of the frequency combinations are of small probabilities. For those combinations in which all the $T-1$ users choose small probability frequencies, the probabilities of the combinations are also very small. Omitting these small probability combinations will not introduce serious errors into the metric. So, one approach for simplifying the channel metric $p(\bar{F}|f_{j_0}^{(0)})$ is to consider only the $K(T)$ most probable frequency combinations in the set C, where $K(T)$ is with polynomial complexity. Denote the set of these $K(T)$ combinations by C', (4.8) becomes

$$p(\bar{F}|f_{j_0}^{(0)}) \approx \sum_{c \in C'} P(c|f_{j_0}^{(0)}) P(\bar{F}|c, f_{j_0}^{(0)}). \tag{4.19}$$

In searching the most probable combinations, if the measurement is given by

$$P_{NSMP} = \left(\frac{1}{N_o}\right)^M \prod_{u=1}^{T-1} p(f_{i_u}^{(u)}) \prod_{j=0}^{M-1} e^{\left(-\frac{(\mu_j\sqrt{E_s}-\sqrt{\rho_j})^2}{N_0}\right)}. \tag{4.20}$$

the approach is called "Narrow-sense most probable combinations" (NSMP). If the measurement is

$$P_{WSMP} = \prod_{u=1}^{T-1} p(f_{i_u}^{(u)}), \tag{4.21}$$

the approach is called "Wide-sense most probable combinations" (WSMP). Generally, it is hard to find the NSMP's except for the brute force check, which requires computation of all the M^{T-1} combinations. There are methods in implementing the WSMP's by the polynomial complexities. However, the *a priori* information $p(f_{i_u}^{(u)})$ is unavailable in the first iteration. Thus, the *a priori* information given by the first iteration is highly suboptimal. And, by exchanging the imprecise *a priori* information, the users introduce interference to each other. This leads to poor performance for later iterations, especially when the system load is high. To fix this problem, NSMP is applied in the first iteration, since it is possible to implement NSMP in the first iteration with low complexity.

In the following discussion, the linear approaches for WSMP is discussed firstly. Then, an NSMP approach for the first iteration is given, which applies the Viterbi algorithm in searching the combinations with highest value for (4.20).

4.4.2 The K-Most Probable Combinations Approach (KMPC)

In the KMPC approach, the set of the most probable frequency combinations, C', is of size K. In C', the most probable combination considered is the one in which all the $T-1$ users choose their most probable frequencies. For other combinations considered, there are $T-2$ users that will choose their most probable frequencies. And, only one user can choose a less important frequency. To decide which user can choose the less important frequency, all $T-1$ users' less important frequencies are queued in a list according to their values. In constructing the first combination other than the most probable one, pick the first frequency from the list, which belongs to one of the $T-1$ users. For this user, the frequency from the list will be chosen and each of the other $T-2$ users will choose the most probable frequencies. For the second combination other than the most probable one, pick the second frequency from the list, which belongs to one of the $T-1$ users. For the other $T-2$ users, the most probable ones will be chosen. Repeat this procedure until all the K frequency combinations are obtained. Generally, the number of combinations K is independent of the user number T, though the value of K should be adjusted when T increases so that the performance of the system does not decrease. Since there are M possible frequencies for each user,

the list of the less important frequencies can have at most $(T-1)(M-1)$ frequencies. So, the maximal value of K in the KMPC approach is

$$K_{max} = 1 + (T-1)(M-1). \tag{4.22}$$

In this case, M of each user's frequencies are considered in constructing the combinations.

A disadvantage for this approach is that, when one of the users has frequency probabilities that are much higher than those of other users, the resulting combinations will bias to information from this user. Hence, the computed metric is biased and will degrade the performance of the consensus decoder.

As an example, an *a priori* probability distribution for the 4-user case is given in Table 4.1. 4 most probable combinations are considered in this example (K is 4). To compute $p(\bar{F}|f_{j_0}^{(0)})$ for user 0, the frequency combinations obtained by using the 4MPC are given in Table 4.2. From these results, it is found that four frequencies of user 1 are chosen while only one frequency is chosen for user 2 and user 3. This is because user 1 puts most of the weights on frequency 0, 1, 3, and 5. As a result, the value of $P(\bar{F}|f^{(0)})$ will bias to the information towards user 1. In other words, user 1 will interfere the detection results of other users.

4.4.3 The k-Most Probable Frequencies Approach (kMPF)

In the kMPF approach, for each of the $T-1$ users, the k most probable frequencies are considered in constructing the combination set C'. The most probable combination considered is the one in which all the $T-1$ users choose their most probable frequencies. For other combinations considered, there are $T-2$ users that will choose their most probable frequencies. For the last user, it is possible for him to choose the second, the third, and up to the k^{th} less probable frequency. So, in the kMPF approach, each of the $T-1$ users has $k-1$ chances to choose a frequency other than the most probable one. And, the relationship between K and T is as follows

$$K(T) = 1 + (T-1)(k-1). \tag{4.23}$$

Table 4.1: An a priori probabilities distribution for the 4-user case.

	$P(f_0)$	$P(f_1)$	$P(f_2)$	$P(f_3)$	$P(f_4)$	$P(f_5)$	$P(f_6)$	$P(f_7)$
U_0	0.2	0.2	0.1	0.1	0.2	0.1	0.1	0
U_1	0.25	0.27	0	0.23	0	0.25	0	0
U_2	0.2	0.1	0.06	0.15	0.13	0.17	0.1	0.1
U_3	0	0.15	0.3	0.22	0.1	0.08	0	0.15

Table 4.2: Combinations considered by the 4MPC and 2MP for user U_0.

	c0			c1			c2			c3		
Users	U_1	U_2	U_3	U_1	U_2	U_3	U_1	U_2	U_3	U_1	U_2	U_3
4MPC	f_1	f_0	f_2	f_0	f_0	f_2	f_5	f_0	f_2	f_3	f_0	f_2
2MP	f_1	f_0	f_2	f_0	f_0	f_2	f_1	f_5	f_2	f_1	f_0	f_3

Note that, when k is equal to the number of frequencies M, the kMPF approach is the same as the KMPC approach where K is equal to K_{max} as computed in (4.22).

The kMPF approach can keep fairness among all users in computing the metric, since k frequencies for each user are considered. So, interference from a particular user will not happen for the kMPF approach. However, the combinations obtained in this way are not necessarily the most probable ones over all the possible combinations in the set C. With the same number of combinations considered, the kMPF approach is not supposed to have a better performance then the KMPC approach.

When computing $p(\bar{F}|f_{j_0}^{(0)})$ for user 0 from the distribution given in Table 4.1, the frequency combinations obtained by 2MPF ($k = 2$) is given in Table 4.2. It can be observed that, though user 1 puts most of the weights on frequency 0, 1, 3 and 5, the results are fair for all the other users. Each user has considered two frequencies in this case.

4.4.4 The Narrow Sense Most Probable (NSMP) Approach

In the first iteration, no *a priori* information is available. Usually, uniform distribution is used for initializing the probabilities $p(f_{i_u}^{(u)})$. Thus, in search-

ing the most probable combinations by the measurement in (4.20), the value of the first product $\prod_{u=1}^{T-1} p(f_{i_u}^{(u)})$ does not make sense. To get NSMP's in the first iteration only needs to find the combinations by the second product

$$P_{dist} = \left(\frac{1}{N_0}\right)^M \prod_{j=0}^{M-1} e^{\left(-\frac{(\mu_j\sqrt{E_s}-\sqrt{\rho_j})^2}{N_0}\right)}, \tag{4.24}$$

which measures the distance between the combination and the received signal. Define the pattern of a frequency combination by the vector $\bar{\mu}$ given in (4.1). If the pattern for combinations with the highest probability product P_{Rice} is known, the number of these combinations can be computed by

$$\begin{aligned} N &= \binom{(T-1)}{\mu_0}\binom{(T-1-\mu_0)}{\mu_1}\cdots\binom{(T-1-\mu_0-\mu_1-\ldots-\mu_{M-1})}{\mu_{M-1}} \\ &= \frac{(T-1)!}{\mu_0!(T-1-\mu_0)!}\frac{(T-1-\mu_0)!}{\mu_1!(T-1-\mu_0-\mu_1)!} \\ &\quad \cdots\frac{(T-1-\mu_0-\mu_1-\ldots-\mu_{M-2})!}{\mu_{M-1}!(T-1-\mu_0-\mu_1-\ldots-\mu_{M-1})!} \\ &= \frac{(T-1)!}{\mu_0!\mu_1!\ldots\mu_{M-1}!}, \end{aligned} \tag{4.25}$$

where $(T-1-\mu_0-\mu_1-\ldots-\mu_{M-1})! = 0! = 1$. If there is some μ_j equals $T-1$ for $0 \le j \le M-1$, then μ_k will be zero for $k \ne j$. So, N is 1 in this case. When the number of frequency M is greater than the number of users T, then it is possible that each user chooses a different frequency. In this case, μ_j is 1 for all $0 \le j \le M-1$, and N is equal to $(T-1)!$. Generally, N has a value between 1 and $(T-1)!$. When the spreading factor is of a moderate value (between 1 and 10), N can be small. In order to explore the

advantage of CFSK in the aspect of channel capacity, practical applications are interested in spreading factors between 1 and 10. At last, the channel transition probability can be computed as

$$P(\bar{F}|f^{(0)}) = NP_{NSMP} = NP_{WSMP}P_{dist}, \tag{4.26}$$

where P_{WSMP} is the same for all combinations in the first iteration.

In order to find the pattern of the combinations with the highest P_{dist}, a procedure that the $T-1$ users make choice from the M frequencies is simulated. Suppose that there are M stages. At the j^{th} stage, there are μ_j users choose the frequency f_j. On each stage, if there are i users have made their choices, then the procedure is at the state S_i. Thus, the procedure will always start from state S_0. After each stage, there are at most $T-1$ users choose the frequency. So, state S_0 can connect to state S_0, state S_1 and up to state S_{T-1}. Similarly, for state S_i, there are at most $T-1-i$ users choose the frequency. Hence, state S_i can connect to state S_i, state S_{i+1} and up to state S_{T-1}. For state S_{T-1}, since all the $T-1$ users have made their choices, no user can choose the frequency in the current stage. So, state S_{T-1} can only connect to state S_{T-1}.

With the states and stages defined above, a trellis can be constructed for describing the frequency choosing procedure. In Figure 4.5, a trellis of 8-user 8-frequency system is described. The purpose is to find the combination

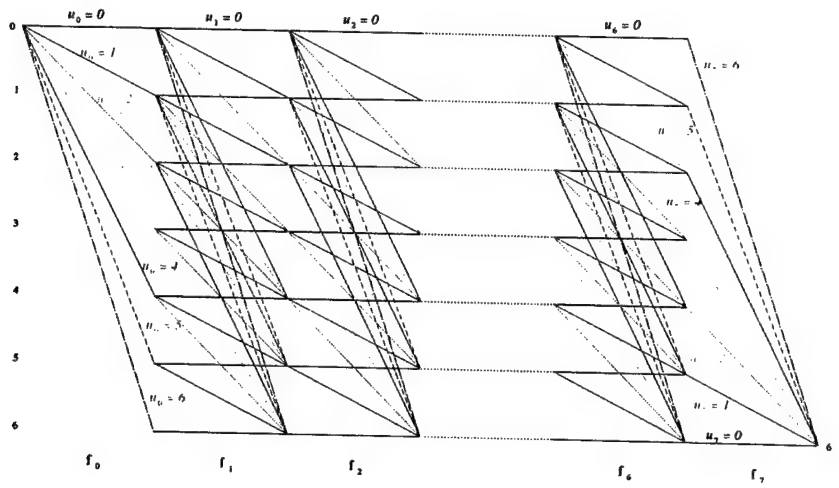

Figure 4.5: A trellis for searching the pattern of combinations with the highest P_{Rice} in a 8-user 8-frequency system.

pattern for computing (4.26). So, we need to consider the choice of U_1, U_2, $\ldots$, U_7. Thus, there are 8 states and 8 stages in Figure 4.5. At the j^{th} stage, if the state is S_i and μ_j users choose frequency f_j, then the next state should be $S_{i+\mu_j}$. The trellis start with state S_0 since no user make choose at first. And, the trellis ends up with state S_{T-1} since all the $T-1$ users are supposed to have made their choices after considering all the M frequencies.

By observing the trellis in Figure 4.5, it is not difficult to find that each combination pattern denoted by $\tilde{\mu}$ can be described by a path through the trellis. The purpose here is to find a path with the highest value for P_{dist}.

P_{dist} can be written in another way as

$$P_{dist} = \left\{ \frac{1}{N_o} e^{\left(-\frac{(\mu_0 \sqrt{E_s} - \sqrt{\rho_0})^2}{N_o} \right)} \right\} \left\{ \frac{1}{N_o} e^{\left\{ \frac{(\mu_1 \sqrt{E_s} - \sqrt{\rho_1})^2}{N_o} \right)} \right\} \ldots \left\{ \frac{1}{N_o} e^{\left(-\frac{(\mu_{M-1} \sqrt{E_s} - \sqrt{\rho_{M-1}})^2}{N_o} \right)} \right\}. \tag{4.27}$$

Thus, there is a metric corresponding to each branch of the trellis. When user 0 does not send frequency f_j, the metric for the branch $\mu_j = \tau$ on the j^{th} stage is computed as

$$m(j, \tau) = \frac{1}{N_o} e^{\left(-\frac{(\tau \sqrt{E_s} - \sqrt{\rho_j})^2}{N_o} \right)}. \tag{4.28}$$

If user 0 send frequency f_j, the metric for the branch $\mu_j = \tau$ on the j^{th} stage is then computed as

$$m(j, \tau) = \frac{1}{N_o} e^{\left(-\frac{((\tau+1) \sqrt{E_s} - \sqrt{\rho_j})^2}{N_o} \right)}. \tag{4.29}$$

Since the metric is defined for each branch of the trellis, it is routine to find a path though the trellis with the highest product P_{dist} in (4.27) by Viterbi algorithm.

Figure 4.6 presents an example of searching procedure on a trellis of a 4-user 8-frequency system. This procedure finds a path through the trellis with the highest P_{dist} so that the channel transition probability $P(\bar{F}|f^{(0)})$ given in (4.26) can be computed.

At every stage of the trellis, the branch metric is computed by (4.29). The partial path product through each input branch is computed at each state.

After comparison at every state, a branch leading to the highest partial path product is survived, and all the other input branches of the state will be deleted. Note that state S_0 has only one input branch at each stage, this branch becomes the survived branch automatically. Similarly, after the first stage, each state has only one input branch. It is kept naturally. On the last stage, the partial path products are computed only at state S_3, and only one branch is survived. Then, a trace back procedure along the survived branch is started from the state S_3 on the last stage, until the state S_0 on the first stage is reached. The path obtained by the trace back procedure is the one with the highest product P_{dist}. Moreover, the μ_j's corresponding to each branch of the path compose to a combination pattern. In Figure 4.6, the obtained pattern is

$$\mu = \{0, 1, 1, 0, 0, 0, 1, 0\}. \tag{4.30}$$

By (4.25), the number of combinations that have the pattern in (4.30) is 6. In the first iteration of the MAP-consensus decoder, the *a priori* probability $p(f_{i_u}^{(u)})$ is initialized by $1/M$, which equals to 0.125 in this example. Thus, P_{WSMP} and $P(\bar{F}|f^{(0)})$ can be obtained by (4.21) and (4.26) respectively.

For the Viterbi algorithm operating on the trellis given Figure 4.5 and 4.6, $i+1$ operations are needed at the state S_i. So, $\sum_{i=1}^{T} i$ operations are needed at each stage. Since there are M stages, the computation needed for the

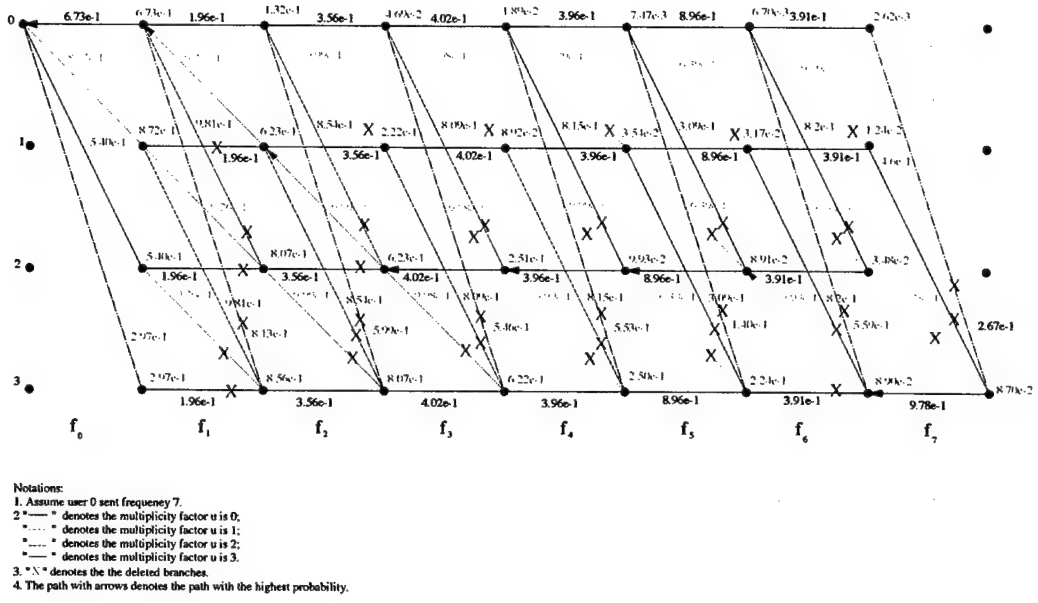

Figure 4.6: An example for the Viterbi algorithm in a 4-user 8-frequency system.

Viterbi algorithm is

$$C_v = M \sum_{i=1}^{T} i = M \frac{T(T+1)}{2}. \tag{4.31}$$

Hence, the computational complexity of Viterbi algorithm is $O(T^2)$, which is a polynomial complexity. This is also the computational complexity for NSMP in the first iteration.

Chapter 5

Performance of the Iterative Detector

5.1 Introduction

Performances of the synchronous and asynchronous systems are constrained by the capacity results given in Chapter 2. When the spreading factor is of moderate and high values, the performances of the synchronous and the asynchronous system are similarly. In this Chapter, performance of the synchronous CFSK system will be discussed basically. In discussing the performance of the simplified metrics, performance of the asynchronous systems will also be given, since we are more interested in the applications of the asynchronous system when the spreading factor is of moderate values. When the spreading factor is very low, i.e. the system load is heavy loaded, results in Chapter 2 display that the capacity of the asynchronous system is much

worse. Especially, when the spreading factor tends to 0, the capacity of the asynchronous system also approaches 0. Thus, we are more interested in application of the synchronous system in this region. Results show that the current system can not achieve decent performance in this region even for the synchronous system, since the error floor is extremely high. Specially code design schemes should be applied for the system. This will be the subject of Chapter 6.

MLSD gives the optimal detection results for the CFSK system. Performance given by iterative detection (implemented by MAP-consensus decoder) is suboptimal. Error floor may exist for both MLSD and iterative detection. When the codes are not uniquely decodable, there is error floor for MLSD and iterative detection. This part of error floor can be very low. For iterative detection, when system load is high, high error floor exists due to the suboptimality of the iterative detection.

When the simplification approaches, such as KMPC and kMPF, are applied in the iterative detection, the error floor is much higher. This is because that the performance of KMPC and kMPF in the first iteration is poor. Using NSMP in the first iteration can improve the performance of simplified iterative detection.

The suboptimality of the iterative detection is discussed in detail. Also, vi-

bration for the convergence of the iterations is analyzed. Possible approaches for solve the error floor problem is discussed. As the first attempt to solve this problem, channel interleavers are applied in the system. The system performance is improved when the system load is not high.

5.2 Comparison of MLSD and Iterative Detection

As introduced in Section 2.4.2, MLSD gives the optimal detection results for the CFSK system. Due to the complexity of MLSD, iterative detection is preferred, which is a suboptimal approach. For application when system load is low, we expect that the iterative detection can approach the performance of MLSD. When, system load is higher, the suboptimality of iterative detection makes the performance much worse than MLSD. To observe the performance of MLSD, only a couple of simple systems can be simulated so that the complexity of MLSD is not forbiddingly high. Thus, we choose the codes given in Figure 2.4 and 2.5 for simulation. Each code block has 250 symbols. 16 iterations are used in the iterative detection.

In Figure 5.1, the average bit error rate (BER) performance for the 2-user synchronous CFSK system is shown. For each user, a rate 1/2 convolutional code is randomly chosen. The number of frequencies, M, is $2^2 = 4$. The

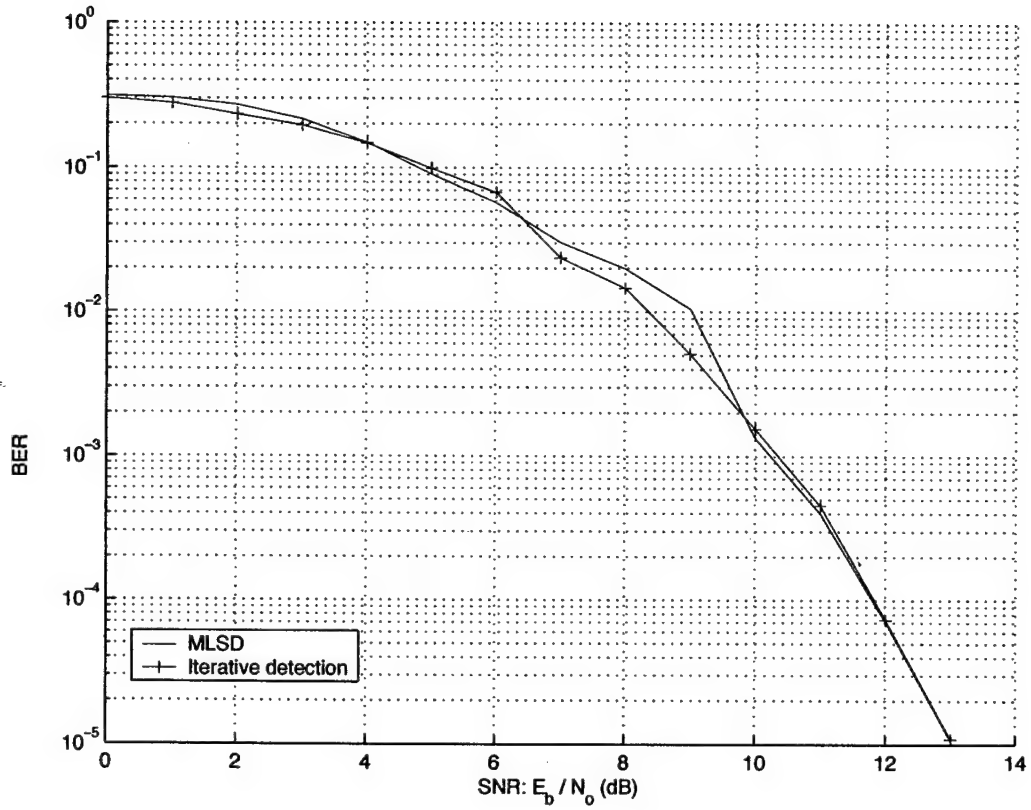

Figure 5.1: The average BER performance of the 2-user system with codes given in Figure 2.4.

code set is given in Figure 2.4. Since the code set is uniquely decodable, no error floor is observed in Figure 5.1 for the BER performance of MLSD. The iterative detection (ID) also provides a decent performance, which approximates to the optimal performance given by the MLSD. The suboptimality of iterative detection does not show up in Figure 5.1, because the system load is not high (spreading factor is 2).

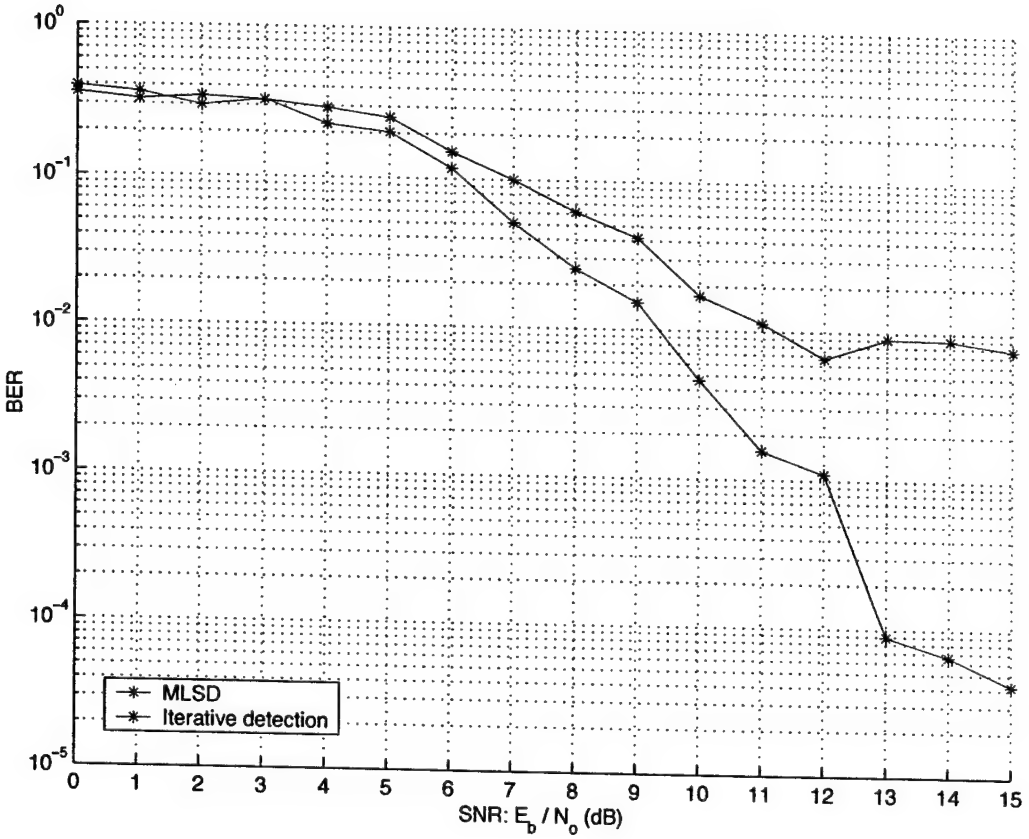

Figure 5.2: The average BER performance of the 3-user system with codes given in Figure 2.5.

The average BER performance for the 3-user synchronous CFSK system is given in Figure 5.2, where the codes are described in Figure 2.5. Since this code set is not uniquely decodable, the BER curve of the MLSD approaches to an error floor when the SNR is high. The level of this error floor is decided by the multiplicity of the paths through the super-trellis with the same output sequence. For the iterative detection, a high level error floor is observed. As discussed before, the non-unique decodablity of the codes contributes to a part of the error floor. The level of this part of error floor equals that of the MLSD. In this case, the system load is higher (spreading factor is 1.33), and the suboptimality of iterative detection makes the error floor much higher.

5.3 Performance of the Simplified Metrics

5.3.1 Applications for Moderate Spreading Factors

When the system load is moderate, the simplified metrics using KMPC or kMPF approach may perform worse. This is because that no *a priori* information is available in the first iteration. However, KMPC and kMPF will depend on the *a priori* information to choose the most probable frequency combinations. The imprecision resulted in the first iteration is exchanged among the users in later iterations. When the system load is moderate or

high, the *a priori* information in the first iteration is much more imprecise, and such imprecision cannot be solved in later iterations. Thus, poor performance is anticipated for KMPC and kMPF approaches.

If NSMP is applied in the first iteration, the performance of the first iteration is supposed to approach that of the optimal metric, i.e. the metric without simplification. Thus, the *a priori* information given by the first iteration is less suboptimal. This makes it easier for later iterations to approach to better performance, though KMPC or kMPF is used in later iterations.

To observe the effects discussed above, the synchronous systems using rate 1/3 codes with memory 6 are simulated. The number of frequencies, M, is 8. The number of users is varied from 1 to 4. The code block length is 250 symbols, and 16 iterations are used. For the simplification approaches, 2 combinations is used for KMPC and kMP when user number is 2. Thus, T is 2 and K is 2. From (4.23), k is 2 for kMPF approach. Discussion in Section 4.4.3 shows that for the 2-user system, KMPC and kMPF are exactly the same when $K = k$. Thus, 2MPC and 2MPF have the same performance. For the 4-user system, 22 combinations are used for KMPC and kMPF. Thus, T is 4 and K is 22. From (4.22), the maximum number of combinations used for KMPC and kMPF, K_{max}, is 22. Also, from (4.23), k is 8 for kMPF approach when K is 22. Since $K = K_{max}$, KMPC and kMPF are exactly

that same. Thus, 22MPC and 8MPF have the same performance for the 4-user system.

The results are given in Figure 5.3. The single user performance provides a lower bound for other systems with the same code rate and same memory. The performance of the 2-user system with the optimal metric is about $0.5dB$ worse than the single user performance. And, the performance of the 4-user system with the optimal metric is nearly $2dB$ worse than the single user performance. No error floor can be observed for the optimal metric performance. But, this does not exclude the possibility that an extremely low level error floor exists since the codes are randomly generated, and the unique decodability is not guaranteed. When 2MPC or 2MPF is used for the 2-user system, the performance is much worse. The BER curve converges to a high error floor. The error floor is due to the high suboptimality of the KMPC and kMPF approaches. When NSMP approach is used in the first iteration, and 2MPC or 2MPF is used in later iterations, the performance is much better, which approximates the performance given by the optimal metric. For the 4-user system, similar results can be observed for 22MPC and 8MPF, which have much higher error floor. When NSMP is applied in the first iteration, the error floor is lowered to a much lower level. However, performance given by $NSMP$ plus 22MPC/8MPF is still more than $1dB$

worse than the performance of the optimal metric. This is because that the spreading factor in this case is 2, which is lower than the 2-user system (spreading factor is 4). The suboptimality for later iteration using 22MPC and 8MPF is too high when spreading factor is 2.

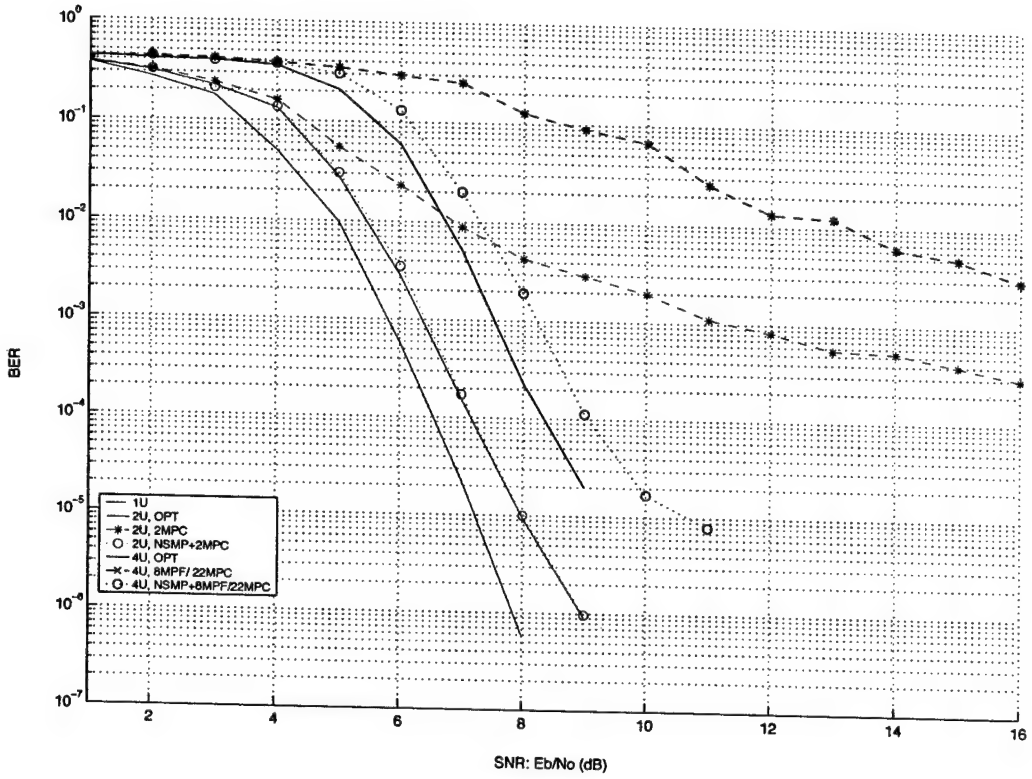

Figure 5.3: The average BER performance of the synchronous 4-user system with rate 1/3 code, memory 6.

Since we are more interested in the asynchronous system in practical applications, we also simulated the asynchronous CFSK system with rate 1/3, memory 6 codes. Results are given in Figure 5.4. The performance given by

the optimal metric is worse than that of the synchronous system. Error floor can be observed for both the 2-user system and the 4-user system. Especially, for the 4-user system, high error floor is observed for the optimal metric performance. Such degradation of performance is decided by the capacity results for asynchronous system given in Chapter 3. However, we can still have the same conclusions for the simplification approaches. When 2MPC/2MPF is used for 2-user system, or 22MPC/8MPF is used for the 4-user system, the error floor is much higher than the one given by the optimal metric. Also, application of NSMP in the first iteration lowers the error floor.

In Figure 5.5, convergence of the asynchronous 4-user system is displayed. For the first iteration, BER of 22MPC/8MPF is much higher than the BER of the optimal metric. The poor performance in the first iteration for 22MPC/8MPF leads to small step size of convergence in later iterations. Thus, the iteration converges to a much higher error level. When NSMP is used in the first iteration, the BER approximated the one given by the optimal metric. Though 22MPC/8MPF is used in later iterations, the step size of convergence is larger. Thus, the curve converges to a lower level.

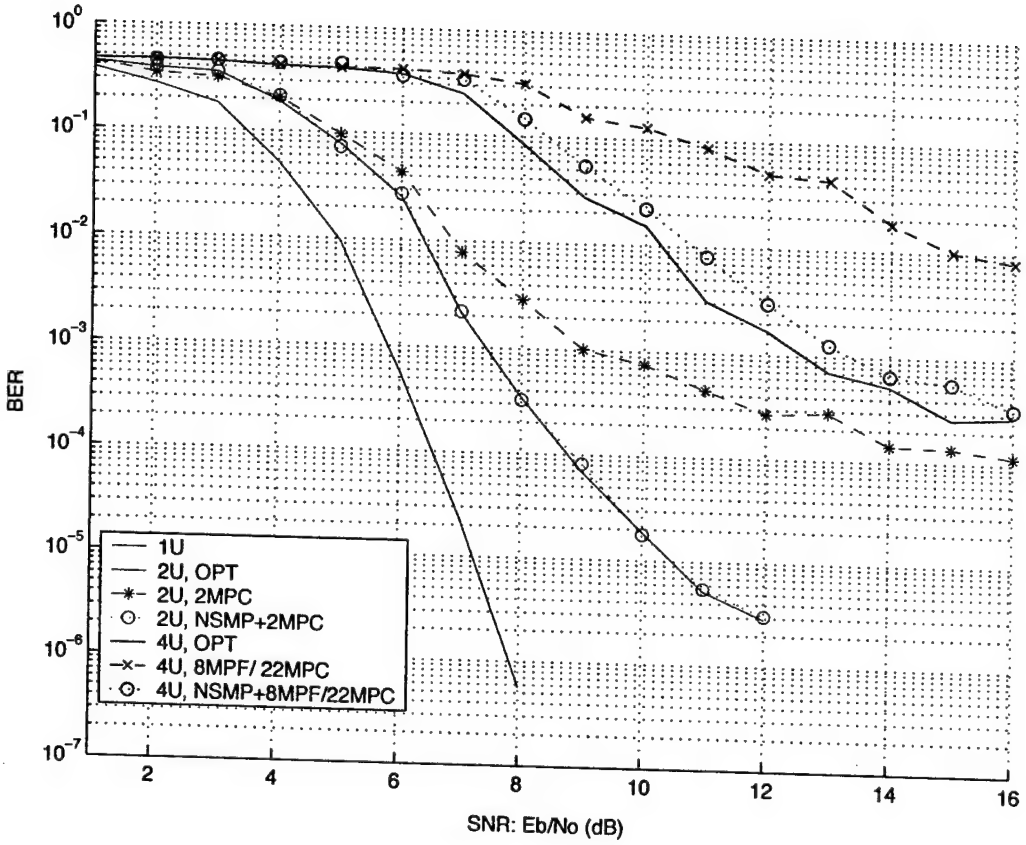

Figure 5.4: The average BER performance of the asynchronous 4-user system with rate 1/3 code, memory 6.

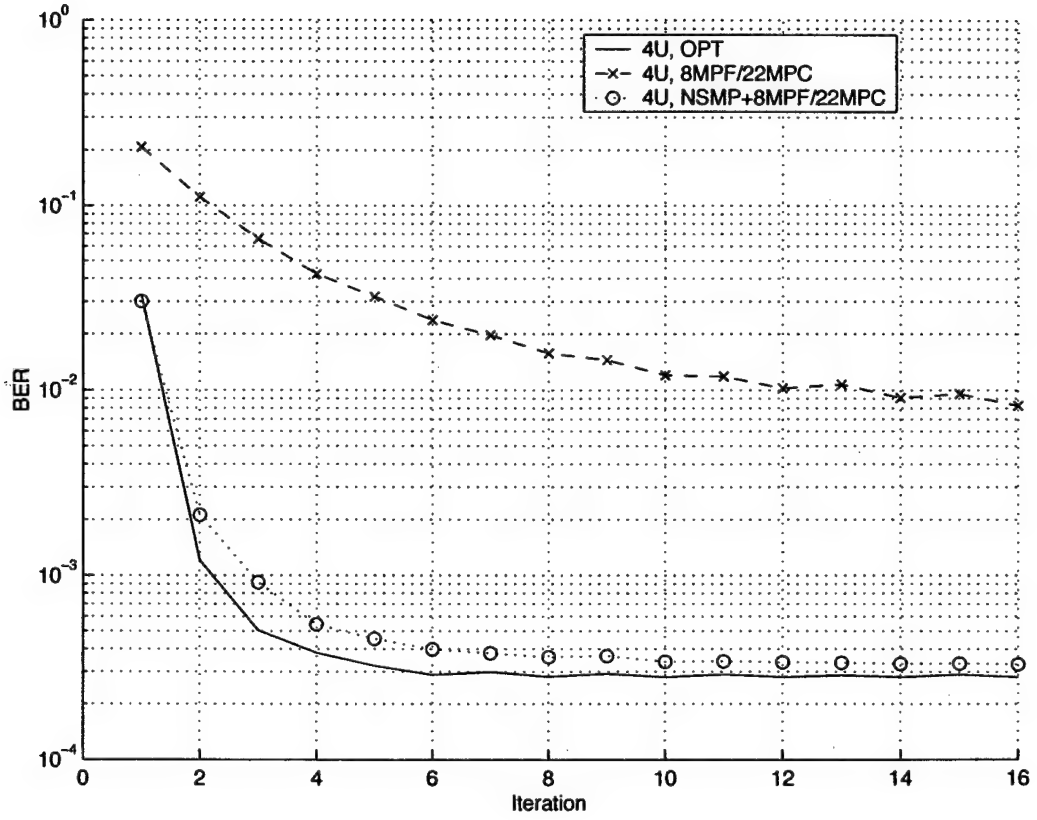

Figure 5.5: The convergence of the asynchronous 4-user system with rate 1/3 code, memory 6.

5.3.2 Applications for High Spreading Factors

To observe the performance of iterative detection when spreading factor is high, more frequencies should be used in the system. Thus, we use rate 1/6 and 2/6, memory 4 codes so that the number of frequencies, M, is $2^6 = 64$. Number of users is varied from 1 to 12. Thus, the spreading factor is varied from 5.3 to 64. From results given in Chapter 3, the synchronous system and the asynchronous system perform similarly. This can be observed in Figure 5.6, where the synchronous 4-user system is compared with the corresponding asynchronous system. Spreading factor is 16 in this case. 4MPC is used for the simulation. When BER is higher than 10^{-3}, the difference between the two systems is smaller than $0.5dB$. When BER is higher than 10^{-4}, the difference is around $1dB$. Though a higher error floor can be observed for the asynchronous system, the difference between the two system will get much smaller when the spreading factor gets higher. Since we are more interested in the application of the asynchronous system when spreading factor is high, later discussions will focus on the asynchronous system. For synchronous system, similar conclusions can be obtained.

For rate 1/6 codes with memory 4, it is too complex to construct the super-trellis for MLSD. Thus, only iterative detection is considered for the high spreading factor applications. For iterative detection, the optimal metric

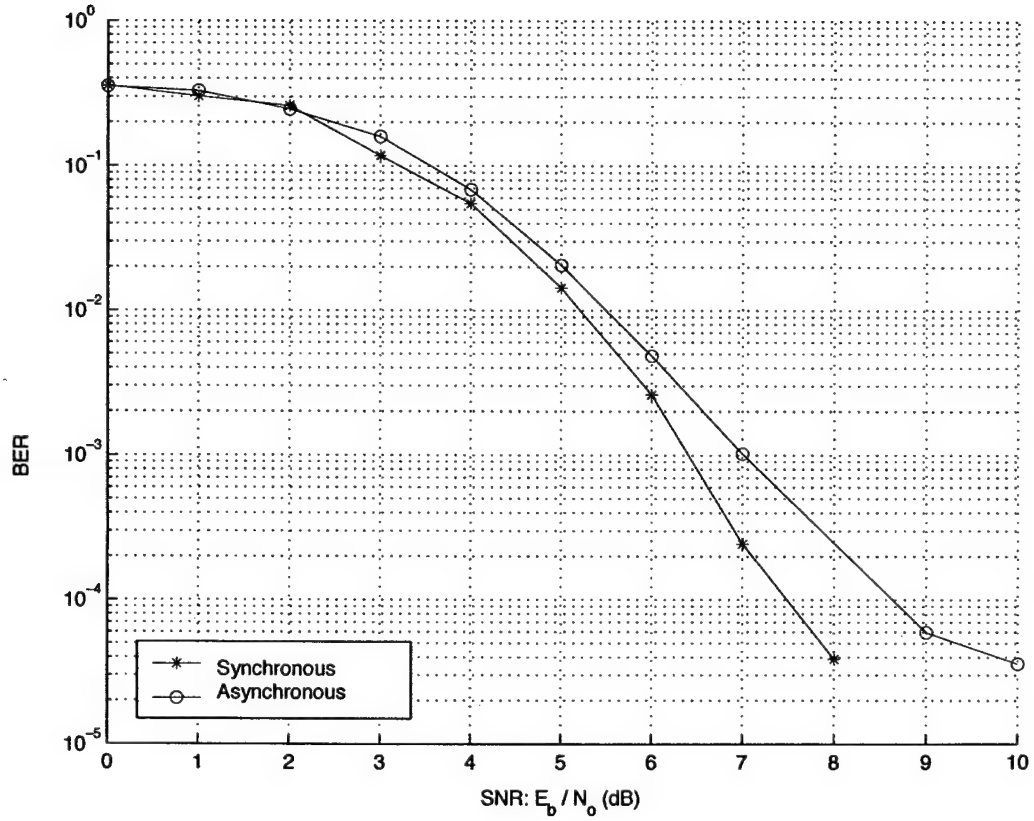

Figure 5.6: Comparison of synchronous and asynchronous 4-user system with rate 1/6 codes, memory 4.

is also too complicated, since the complexity is $O(64^{T-1})$. Thus, it is only possible to use the simplified metrics for the high spreading factor applications. This does not affect later discussions, since the simplified metrics can achieve similar performance as the optimal metric when spreading factor is high. This can be observed in Figure 5.7, where the asynchronous 4-user system with rate 1/6, memory 4 codes are simulated. The results are given by 4MPC, 2MPF and NSMP plus 4MPC. It can be observed that the difference among the three curves is quite small, which is within $0.5dB$. Though NSMP plus 4MPC seems to achieve a lower error floor, the advantage is small. It is reasonable to expect that the performance given by NSMP plus 4MPC can represent the performance given by the optimal metric for the high spreading factor application. Hence, we can conclude that 4MPC and 2MPF can achieve similar performance given by the optimal metric.

In Figure 5.8, performances given by 4MPC for the asynchronous 1-, 2-, 4-, 8- and 12-user system with memory 4, rate 1/6 codes are given. It can be observed that the 12-user system has a high error floor, which is around 4×10^{-4}. For the other 4 systems, the error floors are all lower than 10^{-4}. Such performance is decent for the asynchronous system with 4MPC approach.

For the results given in Figure 5.8, when there are 12 users, the spreading factor is 5.3, and the sum rate of the system is 0.1875 (bits per channel

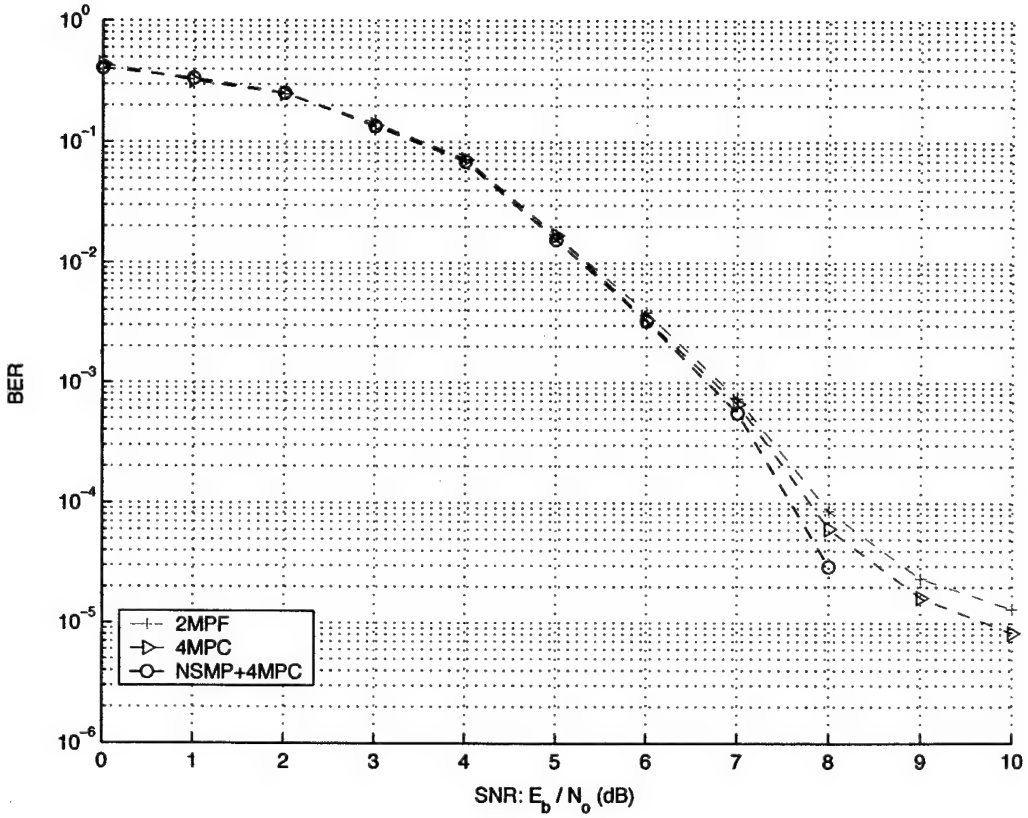

Figure 5.7: Performance of asynchronous 4-user system with rate 1/6 codes, memory 4.

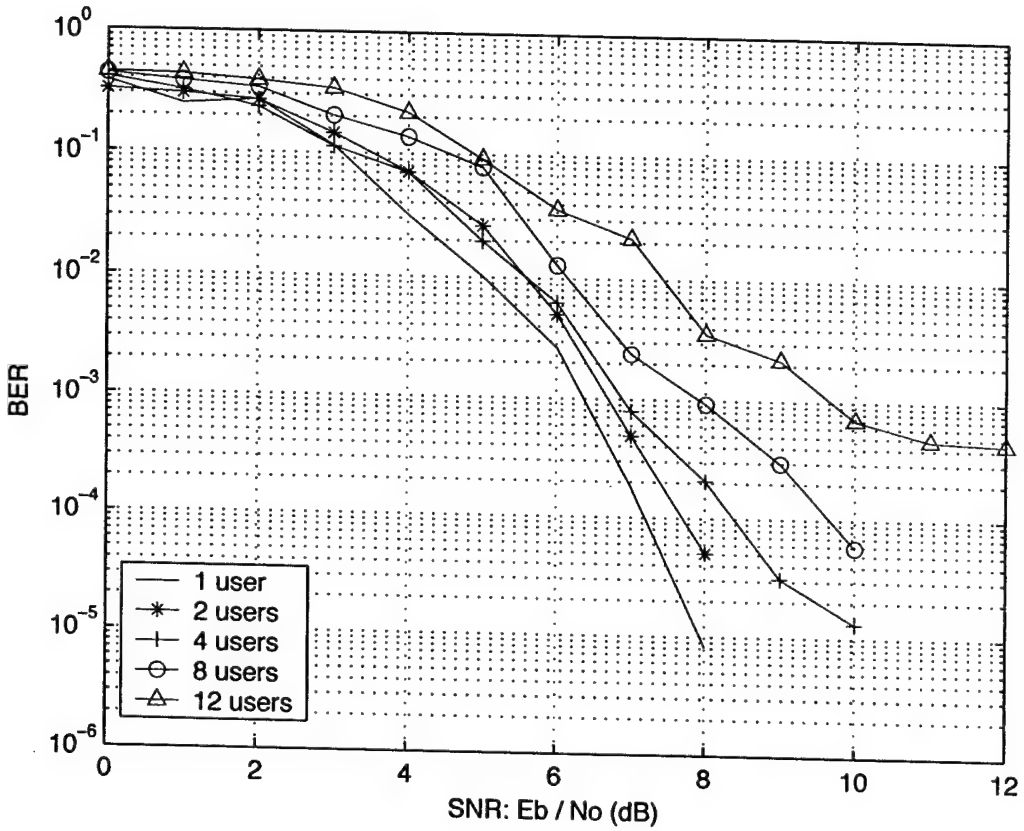

Figure 5.8: Performance of asynchronous CFSK system with 1, 2, 4, 8 and 12 users, using rate 1/6 codes, memory 4.

use per chip). However, the channel capacity of the synchronous system is around 0.8 (bits per channel use per chip) when spreading factor is 5.3. From results given in Chapter 3, the asynchronous system has similar channel capacity for this spreading factor. Thus, the achieved sum rate is only 23% of the channel capacity. To make use of the channel more efficiently, higher code rate should be applied. Thus, the systems with rate 2/6 and memory 4 codes are simulated. The results are given in Figure 5.9. The performances for these systems are worse. When user number is 8, the error floor is around 2×10^{-4}, while no error floor can be observed for the system with rate 1/6 codes. When user number is 2 and 4, the performance is still decent. For the 8 user system, spreading factor is 8 and the channel capacity is around 0.5. The achieved sum rate is 0.25 (bits per channel use per chip) for the 8-user system with rate 2/6 codes, which is 50% of the channel capacity. Thus, the application of the channel is much more efficient. When user number is 12, the performance of this system is much worse than the result given in Figure 5.8, which is ignored in Figure 5.9.

5.4 Suboptimality of Iterative Detection

The suboptimality of iterative detection may lead to high error floor when spreading factor is high. If KMPC or kMPF is applied, the iterative detec-

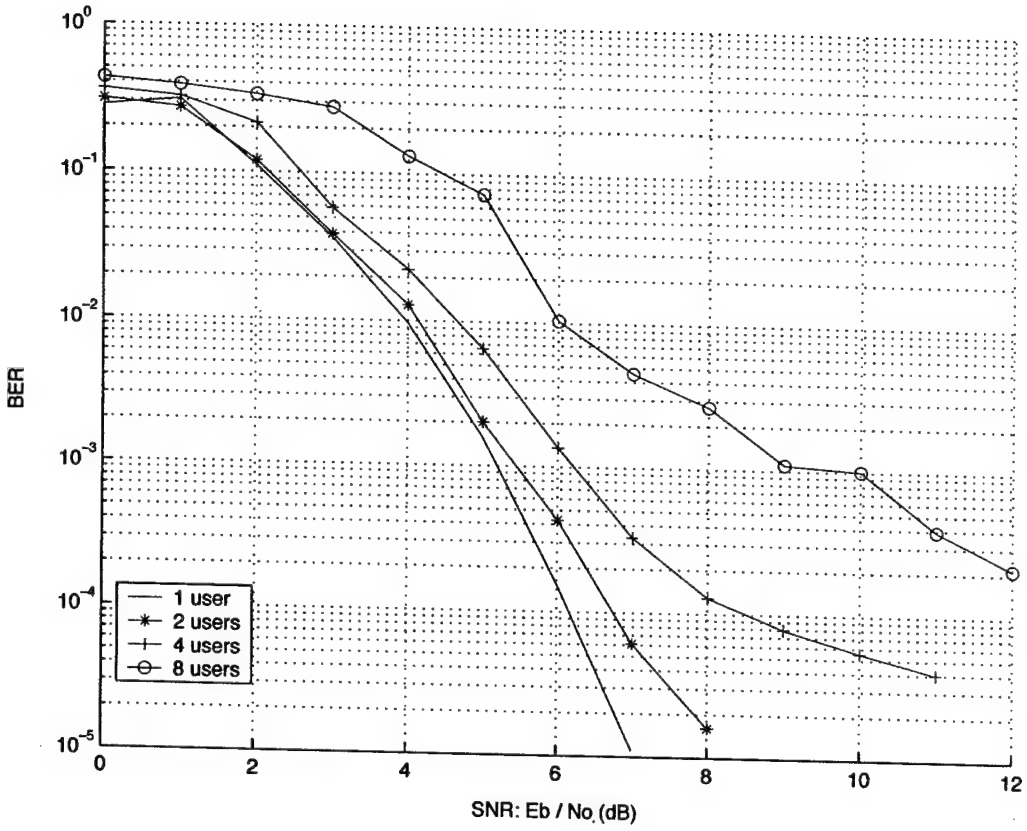

Figure 5.9: Performance of asynchronous CFSK system with 1, 2, 4 and 8 users, using rate 2/6 codes, memory 4.

tion is more suboptimal. Thus, the study of the suboptimality of iterative detection is significant in improving the system performance for high spreading factor applications.

At first, the 2-user system is considered, which is the simplest case. In the very beginning, no *a prior* information is available. It is highly probable that the decisions made by the two users' decoders are not correct. Thus, the computed *a posteriori* information for the symbols is imprecise. When the information is exchanged to the other user, it is equivalent to introducing the multiuser interference (MUI) to the other user.

For the next iteration, whenever the component decoder cannot get over the MUI, new errors are generated in computing the *a posteriori* information. So, the MUI still exists and is delivered to the other user again.

If the MUI is not resolved after many iterations, the error floor of the iterative detection is formed. There are four cases that the MUI can influence the error floor:

1. If both component decoders are powerful, the MUI can be resolved by the decoders. Then, there is no error floor caused by the iterative detection.

2. If one of the component decoders is powerful, it can resolve the MUI

whenever it gets the MUI. However, the other decoder cannot resolve the MUI. The error floor will get lower and lower, and disappear finally. Such convergence is slow since only one component decoder work properly.

3. If one of the component decoders is powerful and can resolve the MUI it obtains. The other component decoder is not powerful, and sends the MUI to the powerful component decoder. However, the MUI is too high, and the powerful component decoder is not strong enough to get rid of the MUI. Thus, such MUI is delivered between the two decoders. This will lead to the vibration of the convergence. Such vibration can be observed in Figure 5.5 for the convergence of 8MPC after the 10 iterations.

4. If both component decoders are not powerful, the MUI is exchanged between the two component decoders. The error floor is high in this case.

Several factors can affect the level of the MUI:

1. In the first iteration, the decision is not precise. MUI are generated. If correlation of the two users' codes is strong, the MUI will be much more serious.

2. In later iterations, the correlation of the users' codes will still affect the level of the MUI.

3. For all iterations, the error correcting ability of the codes will affect the level of the MUI. If the codes are powerful, the MUI are resolved. Otherwise, the MUI are kept or may be more serious.

4. If the simplified metrics are used, the MUI will be more serious or new MUI is generated.

For the system with more than two users, the exchange of MUI is much more complicated. But, similar relationship can be obtained between MUI and error floor.

Though the convolutional codes applied in the previous discussions can be very powerful, when the coded sequence are mapped to the M-ary symbols, the Hamming distance of the codes is decreased. For example, the 4-tuple 1100 are mapped to symbol sequence 30 (4 frequencies used), and 0011 are mapped to 03. The Hamming distance of the bit sequence is 4, the Hamming distance of the 4-ary symbol sequence is only 2. Thus, the signal mapping in this way may degrade the code performance a lot. This is the reason that high error floor is formed when there are too many users.

To reduce the suboptimality of iterative detection, there are 3 possible ap-

proaches:

1. Decrease the correlation of the codes by using the channel interleavers. When there are many users in the system, the MUI is too high after the first iteration. Thus, the channel interleavers can improve the system performance only when spreading factor is of moderate or high values.

2. In order that the MUI is not serious after the first iteration if KMPC or kMPF is used, NSMP approach should be applied in the first iteration.

3. By using non-binary codes, such as the Reed-Solomon codes or convolutional code over $GF(q)$, the distance property of the codes can be guaranteed. Thus, the codes are more powerful in decreasing the MUI level.

For the above three approaches, NSMP is already discussed in Section 4.4.4. The application of channel interleavers will be discussed in the following section. Code design issues will be given in Chapter 6.

5.5 Effects of Channel Interleavers

The channel interleaver is symbol based, which permutes each n-tuple coded bits randomly. As denoted in Figure 2.1, the channel interleaver is concatenated after the encoder at the transmitter. At the consensus decoder,

the users' component decoders also need modifications, as denoted in Figure 5.10. Thus, each user's component needs to know the other $T-1$ users' interleaver structures. This makes the structure of the component decoder more complicated. However, the computational complexity of the decoding algorithm does not increase.

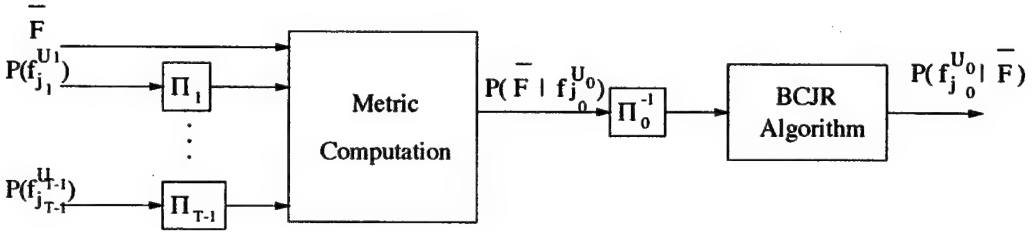

Figure 5.10: The user's component decoder with channel interleavers.

The interleaver size can affect the performance of the detection. Generally, the interleaver size equals the length of the coding block. Figure 5.11 shows the performance of the synchronous 3-user system with codes given in Figure 2.5, and with different interleaver sizes. These results are compared with those given in Figure 5.2. We can observe that when interleaver size increases, the system performance increases. This is because that the random channel interleaver reduces the level of correlation among the users' codes. When the interleaver size is large enough, the users' code will be independent with each other. In such condition, the increase of interleaver size will not improve the performance any more. This can be observed in Figure 5.11 that

when the interleaver size increases from 1000 to 10000, the performance only improves a little, compared with the large improvement when the interleaver size increases from 250 to 1000. Another effect is that the performance of the system with channel interleavers is even better than the performance given by MLSD when the interleaver size is large enough. This is not surprising, since MLSD is without channel interleavers, and the interleavers can increase the memory of the super-trellis. Thus, such comparison is not fair for MLSD.

Since the channel interleavers can improve the system performance, it may be interesting to observe the effects when channel interleavers are applied for 4-user system with rate 1/3 codes. As given in Figure 5.3 and 5.4, poor performance is obtained when 4MPC or 2MPF is used. When NSMP is used for the first iteration, the performance is still worse than that of the optimal metric. After utilizing the channel interleavers in this system, the results are compared with those given in Figure 5.4. In Figure 5.12, it can be found that the performances of 4MPC/2MPF and NSMP plus 4MPC are improved when channel interleavers are used. For high SNR's, their performances approximate the one given by the optimal metric with channel interleavers. And, the error floor with channel interleavers is lower than the error floor without channel interleavers. In Figure 5.13, convergence behavior of the system with channel interleavers is described. When channel interleavers

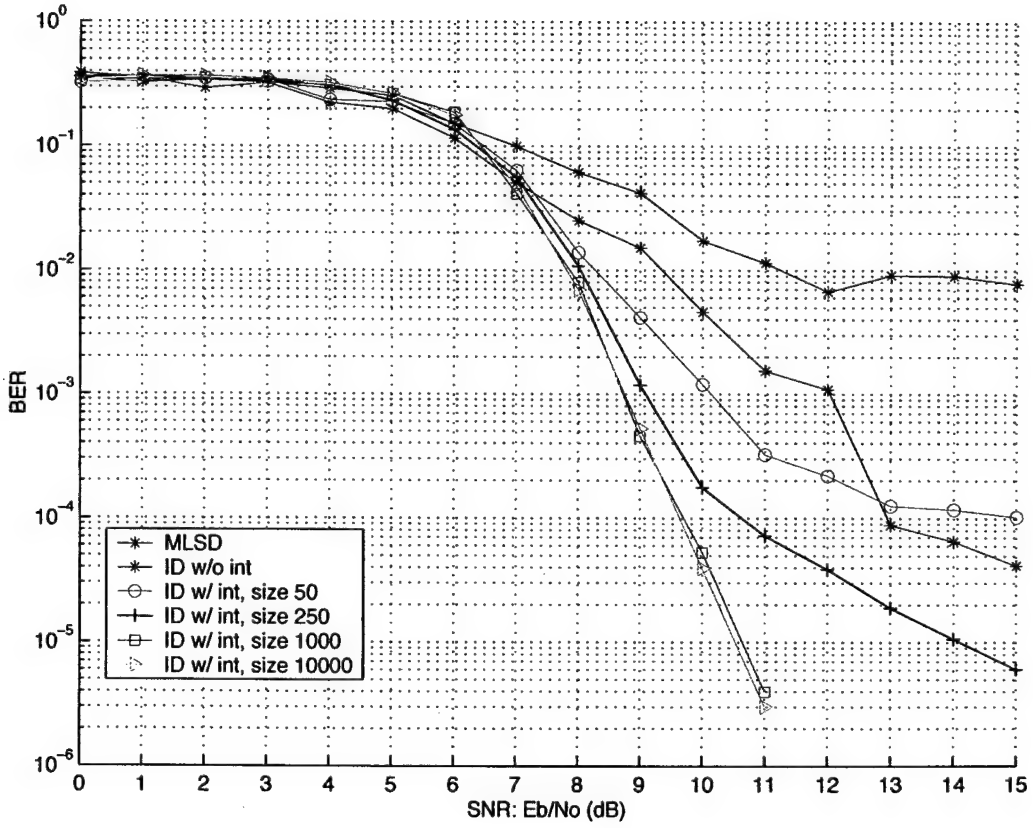

Figure 5.11: Performance of synchronous 3-user system with channel interleavers.(the codes are given in Figure 2.5.)

are applied with 4MPC, the performance of the first iteration is still worse, since the interleavers do not help provide *a priori* information in the first iteration. However, the interleavers make the users' codes less correlated with each other. This help reduce the MUI among users. Thus, it is easier for the component decoders to reduce the MUI in later iterations. In Figure 5.13, we can find that the convergence curve for 4MPC with interleavers goes down with a larger step size than that of the curve for 4MPC without interleavers. Such effect can also be observed for the curves for the NSMP plus 4MPC and the optimal metric. Thus, the whole procedure converges to a low BER level. And, this make NSMP plus 4MPC performs more similar to the optimal metric.

5.6 Summary

In this Chapter, the performance of iterative detection is discussed, where iterative detection is implemented by MAP-consensus decoder. For Turbo-consensus decoder, similar conclusions can be obtained. The iterative detection is suboptimal when compared with MLSD, which provides the optimal performance for multiuser detection of the CFSK channel. The suboptimality of iterative detection is because that the multiuser interference (MUI) is exchanged among users, and the users' component decoders may not be

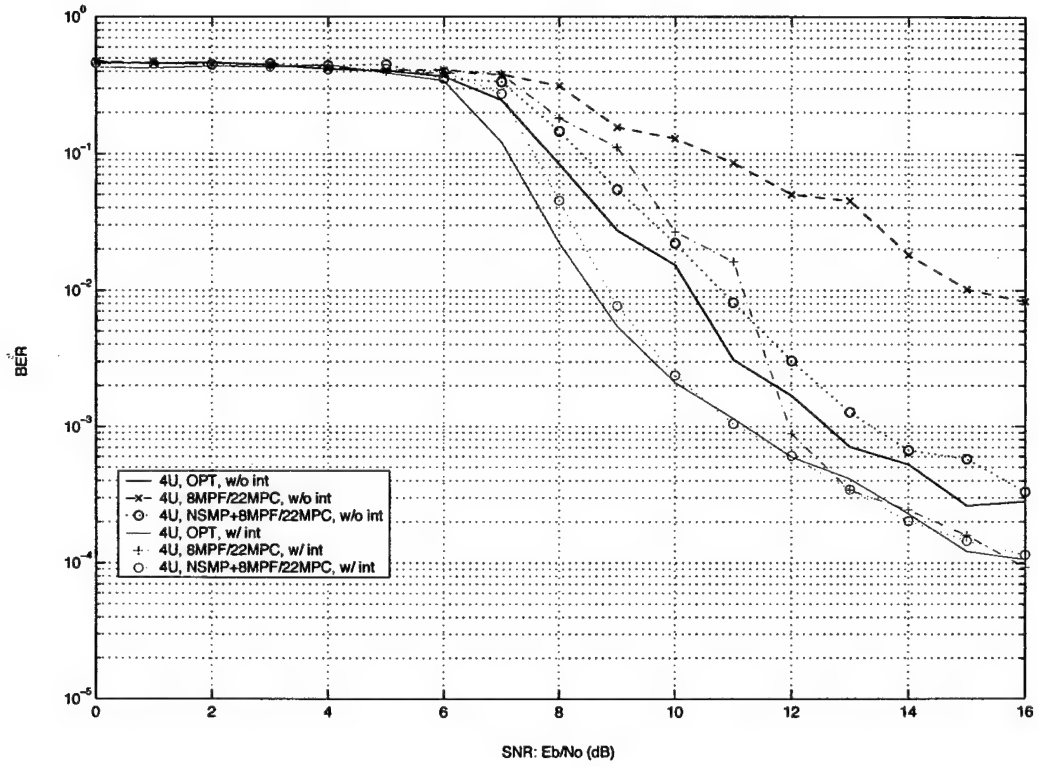

Figure 5.12: Performance of asynchronous 4-user system with channel interleavers, code rate 1/3, memory 6.

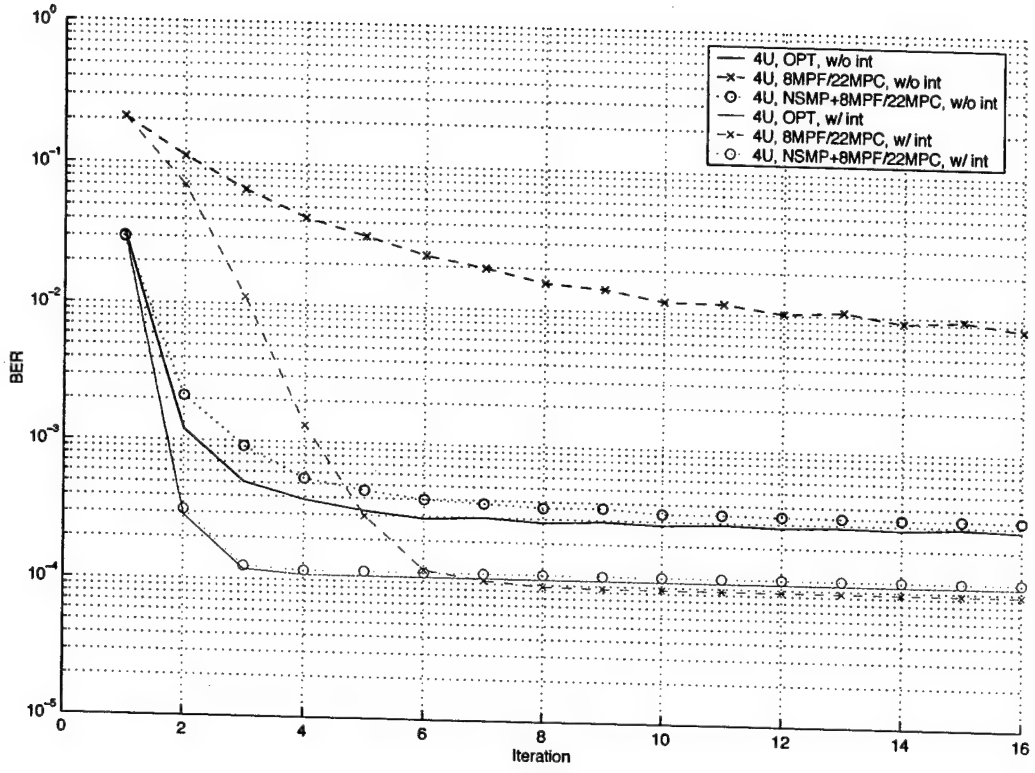

Figure 5.13: Convergence of asynchronous 4-user system with channel interleavers, code rate 1/3, memory 6.

powerful enough to resolve the MUI. Thus, error floor is resulted when MUI cannot be reduced by the users' component decoders.

To lower the level of the error floor, possible approaches include the application of NSMP in the first iteration, the application of channel interleaver and the application of powerful non-binary code for the users' codes. The results have shown that NSMP can improve the performance of KMPC or kMPF effectively. However, such improvement is limited, especially when the system load is high. Channel interleavers can help improve the performance further. For moderate spreading factors, channel interleavers make the simplified metrics work as well as the optimal metric. Also, the error floor of the performance is lowered. However the help provided by channel interleavers is limited, since the ability to reduce the MUI is finally decided by the error correction ability of the channel codes. When spreading factor is low, weak codes cannot resolve the MUI problem at all. In such condition, the application of channel interleavers does not make sense.

An important conclusion is that the key factor of lowering the error floor is the code design. In previous discussion, we find that the randomly generated binary convolutional codes do not provide decent performance since the signal mapping decreases the Hamming distance of the codes. Thus, codes in the $GF(q)$ should be considered. This will be the subject of the following

Chapter.

Chapter 6

Code Design for the CFSK System

6.1 Introduction

In former discussions, the channel codes are randomly generated RSC. For the signal mapping scheme, each n-tuple coded bits are mapped to a frequency. As a result, the symbol sequence can be considered as a randomly chosen non-binary trellis coded sequence. Generally, the unique decodability of the non-binary code is not guaranteed. For applications of low spreading factors, the number of users is high and the problem of uniquely decodable code becomes much more serious. Code design for the non-binary trellis codes with unique decodability is extremely difficult. Known uniquely decodable block codes (UDBC) provide us solution for the design of uniquely decodable non-binary trellis codes [17][37][38][39][28]. In [17], Chang and

Wolf proposed the uniquely decodable block codes with sum rates approaching the capacity bounds for T-user M-frequency channel with and without intensity information. In [37], Kautz and Singleton found the KS code that is based on a maximum-distance-separable (MDS) outer code (e.g., Reed-Solomon code) and an orthogonal weight-one inner code. The KS code is proved to be uniquely decodable on the binary adder channel. The orthogonal weight-one inner code of KS code is similar to the signal mapping function in the CFSK system. In [39], example of the application of KS code based on Reed-Solomon code over $GF(7)$ is discussed. This provides us a clue for applying the Reed-Soloman codes in CFSK system for signal mapping. Note that application of the KS code in CFSK will lead to low code rate, which is far from the capacity bounds given in Chapter 3. A major problem for the above UDBC is that these codes are weak in the error correction capability. A possible approach to solve this problem is proposed in [28], where the UDBC can be concatenated with the binary convolutional code so that decent performance and unique decodability is provided. In Section 6.2, the application of UDBC given in [17] are discussed. Results of the application is given in Section 6.3.

6.2 Application of Uniquely Decodable Block Codes(UDBC)

6.2.1 UDBCs with Sum Rate Approaching the Capacity Bound

In [27], Chang and Weldon proposed an iterative construction for the UDBC on the binary memoryless T-user adder channel. Chang and Wolf extended the construction to non-binary UDBC on the T-user M-frequency channel with intensity information in [17].

The iterative construction works on the difference matrix D_j in the j^{th} iteration. The difference matrix is initialized by $D_0 = [1]$. Then, for the j^{th} iteration, the difference matrix is given by

$$D_j = \begin{bmatrix} D_{j-1} & D_{j-1} \\ D_{j-1} & -D_{j-1} \\ I_{j-1} & \mathbf{0} \end{bmatrix}, \tag{6.1}$$

where D_{j-1} is the difference matrix in the $j-1^{th}$ iteration, I_{j-1} is the identity matrix, and $\mathbf{0}$ is the zero matrix. By the above iterative procedure, D_j has N_j columns ($N_j = 2^j$) and P_j rows ($P_j = (j+2)2^{j-1}$). For a set of codes constructed by j iterations, let $\tilde{D} = D_j$, then the difference matrix for

constructing the codes is given by

$$\tilde{D}^* = \begin{bmatrix} \tilde{D} \\ a\tilde{D} \\ a^2\tilde{D} \\ \vdots \\ a^{M-2}\tilde{D} \end{bmatrix}, \tag{6.2}$$

where a is an arbitrarily chosen integer that satisfies

$$a \geq P_j(M-1)+1. \tag{6.3}$$

Thus, there are

$$T = (M-1)(j+2)2^{j-1} \tag{6.4}$$

rows in the matrix $\tilde{D}^*$. Then, each row of $\tilde{D}^*$ can construct a code for a particular user. Hence, a set of codes for T users can by constructed. The i^{th} code has $y = 2$ codewords, denoted by w_i and w_i'. Denote the k^{th} position in w_i and w_i' by $w_{i,k}$ and $w_{i,k}'$. Then, $w_{i,k}$ and $w_{i,k}'$ can be constructed from the $(i,k)^{th}$ position of $\tilde{D}^*$, $d_{i,k}$, as

$$\begin{cases} w_{i,k} = a^l, \ w_{i,k}' = 0, & \text{if } d_{i,k} = a^l, \\ w_{i,k} = 0, \ w_{i,k}' = a^l, & \text{if } d_{i,k} = -a^l, \\ w_{i,k} = a^{l_1}, \ w_{i,k}' = a^{l_2}, & \text{if } d_{i,k} = a^{l_1} - a^{l_2}. \end{cases} \tag{6.5}$$

Chang and Wolf had proved in [17] that the above construction gives us a set of uniquely decodable codes for T users M frequencies. For the T users, the sum rate can be given by

$$\begin{aligned} R_{sum} &= \frac{Tlog_2 y}{N_j} \\ &= \frac{(M-1)(j+2)2^{j-1}}{2^j} \\ &= \frac{M-1}{2}(j+2), \end{aligned} \tag{6.6}$$

where y denotes the number of codewords in the i^{th} code, and j denotes the number of iterations used. It is not difficult to show that when j tends to infinity, the sum rate R_{sum} will approach the capacity bound of the CFSK channel with intensity information. The sum rates of some constructed UDBC are compared with the capacity bounds of the CFSK channel with intensity information. The results are given in Table 6.1.

Table 6.1: Comparison between sum rates of Chang and Wolf's UDBC and capacity results.

M	T	j	N	R_{sum}	percent of capacity
3	6	1	2	3	69.74 %
3	16	2	4	4	69.97 %
3	224	5	32	7	73.50 %
3	12,288	10	1,024	12	78.42 %
3	23,068,672	20	1,048,576	22	84.05 %
4	18,432	10	1,024	18	76.94 %
4	34,603,008	20	1,048,576	33	83.11 %

For the 6-user 3-frequency case, 1 iteration is used. Thus, $\tilde{D}$ can be constructed from $D_0 = [1]$ as

$$\tilde{D} = D_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & 0 \end{bmatrix}. \tag{6.7}$$

Let the integer a be

$$a = P_1(M-1)+1 = 7, \tag{6.8}$$

where P_1 equals 3. Thus, the difference matrix $\tilde{D}^*$ is

$$\tilde{D}^* = \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & 0 \\ 7 & 7 \\ 7 & -7 \\ 7 & 0 \end{bmatrix}. \tag{6.9}$$

From $\tilde{D}^*$, the code set for six users is constructed as

$$\left\{ \begin{array}{rcl} C_0 & = & \{(1,1),(0,0)\}, \\ C_1 & = & \{(1,0),(0,1)\}, \\ C_2 & = & \{(1,0),(0,0)\}, \\ C_3 & = & \{(7,7),(0,0)\}, \\ C_4 & = & \{(7,0),(0,7)\}, \\ C_5 & = & \{(7,0),(0,0)\} \end{array} \right\}. \tag{6.10}$$

To apply this code set in the CFSK system, we can use the following map

$$\left\{ \begin{array}{rcl} 0 & \rightarrow & f_0, \\ 1 & \rightarrow & f_1, \\ 7 & \rightarrow & f_2 \end{array} \right\}. \tag{6.11}$$

As a result, we obtain the non-binary UDBC for the CFSK channel with intensity information as

$$\left\{ \begin{array}{rcl} C_0 & = & \{(f_1,f_1),(f_0,f_0)\}, \\ C_1 & = & \{(f_1,f_0),(f_0,f_1)\}, \\ C_2 & = & \{(f_1,f_0),(f_0,f_0)\}, \\ C_3 & = & \{(f_2,f_2),(f_0,f_0)\}, \\ C_4 & = & \{(f_2,f_0),(f_0,f_2)\}, \\ C_5 & = & \{(f_2,f_0),(f_0,f_0)\} \end{array} \right\}. \tag{6.12}$$

6.2.2 Concatenation with Convolutional Codes

It is easy to prove that the code set given in (6.12) does not have decent error correction performance. To endow the system with strong error correction

capability, it is necessary to use error correction code with better performance before applying the UDBC. Thus, convolutional codes are utilized in the system before signal mapping (using the UDBC). The transmitter described in Figure 2.1 is modified as described in Figure 6.1. As discussed previously, the symbol-based interleaver is optional. The coded sequence output by the convolution code is then encoded by the UDBC. Since there are two codewords in each user's code, each codeword represents 1 bit. Thus, each bit of the coded sequence output by the convolutional code is mapped to the a codeword in the UDBC set. In Figure 6.1, an example is given by the code C_0 for user U_0. The encoding rule for UDBC is that 0 is mapped to the codeword $(1, 1)$ and 1 is mapped to the codeword $(0, 0)$. For the M-FSK modulation, the symbol 1 is than mapped to frequency f_1 and 0 is mapped to frequency f_0.

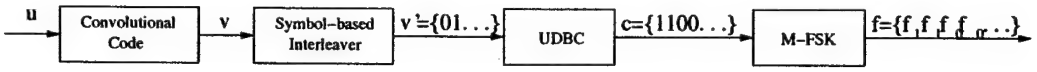

Figure 6.1: Convolutional code concatenated with UDBC in CFSK transmitter.

At the receiver side, the user's component decoder has the same structure as we have discussed in Chapter 4. In the MAP algorithm, the symbol is defined by the branch output on the convolutional code's trellis. In other words, the n-bit symbol can be represented by n codewords of the UDBC.

Computation and decision for the MAP algorithm is made on the symbols, since we need the *a posteriori* probabilities of the symbols. Furthermore, information bits can be obtained from decisions of the symbols, since RSC is used for convolutional code in the transmitter. Note that UDBC endows the trellis code with unique decodability, we can use the same RSC for all users. For the (n,k) RSC, each n-tuple of the coded bits is mapped to a symbol. Thus, there are $N_s = 2^n$ symbols. Each symbol of user U_u is denoted by

$$s_{i_u}^{(u)} = \{f_{j_{1,1}}, f_{j_{2,1}}, \ldots, f_{j_{N,1}}, \ldots, f_{j_{1,n}}, f_{j_{2,n}}, \ldots, f_{j_{N,n}}\}. \tag{6.13}$$

To implement the MAP algorithm, the channel metric needs to be modified as

$$\begin{aligned} p(\mathbf{F}|s_{i_0}^{(0)}) \;=\; & \left(\tfrac{1}{N_o}\right)^{NM} \sum_{i_1=0}^{N_s-1} \sum_{i_2=0}^{N_s-1} \cdots \sum_{i_{T-1}=0}^{N_s-1} \prod_{u=1}^{T-1} p(s_{i_u}^{(u)}) \prod_{j=0}^{M-1} e^{\left(-\frac{(\mu_{j,1,1}\sqrt{E_s}-\sqrt{\rho_{j,1,1}})^2}{N_o}\right)} \\ & \prod_{j=0}^{M-1} e^{\left(-\frac{(\mu_{j,2,1}\sqrt{E_s}-\sqrt{\rho_{j,2,1}})^2}{N_o}\right)} \cdots \prod_{j=0}^{M-1} e^{\left(-\frac{(\mu_{j,N,n}\sqrt{E_s}-\sqrt{\rho_{j,N,n}})^2}{N_o}\right)}, \end{aligned} \tag{6.14}$$

where N denotes the length of the UDBC codeword, $\mu_{j,k,x}$ denotes the number of transmissions on frequency f_j in the k^{th} position of the UDBC codeword for the x^{th} bit of the symbol. The received signal $\mathbf{F}$ is a $M \times N$ matrix

$$\mathbf{F} = \begin{bmatrix} \rho_{0,1,1}, & \rho_{0,2,1}, & \cdots & \rho_{0,N,1} & \cdots & \rho_{0,1,n}, & \rho_{0,2,n}, & \cdots & \rho_{0,N,n} \\ \rho_{1,1,1}, & \rho_{1,2,1}, & \cdots & \rho_{1,N,1} & \cdots & \rho_{1,1,n}, & \rho_{1,2,n}, & \cdots & \rho_{1,N,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \rho_{M-1,1,1}, & \rho_{M-1,2,1}, & \cdots & \rho_{M-1,N,1} & \cdots & \rho_{M-1,1,n}, & \rho_{M-1,2,n}, & \cdots & \rho_{M-1,N,n} \end{bmatrix}. \tag{6.15}$$

in which the element $\rho_{j,k,x}$ denotes the received energy level of frequency f_j in the k^{th} position of the UDBC codeword for the x^{th} bit of the symbol.

Base on the symbol $s_{i_u}^{(u)}$, the structure of the MAP-consensus decoder or the Turbo-consensus decoder is preserved, and the simplification approaches for the metric given in Chapter 4 are also effective.

6.2.3 Performance Results

To observe the performance of the CFSK system using UDBC, the concatenation scheme given in previous section is adopted. 3 frequencies are used in simulations. 1 or 2 iterations are applied in generating the UDBC code set. As given in Table 6.1, the maximum number of users in these cases is 6 or 16. Then, the spreading factors of applications for these cases are generally less then 1. From results in Chapter 3, the capacity of asynchronous CFSK channel with intensity information decreases a lot when spreading factor is less than 1, which implies that only synchronous system is applicable for achieving high coding rate. Thus, synchronous system is used in the simulations. For the convolution code, rate 1/2 RSC is used. Thus, the sum rates achieved by the system is 1/2 of the sum rates given in Table 6.1. The length of the coding block is 250 symbols, Both channel interleavers and NSMP+KMPC are applied for the system.

Results are given in Figure 6.2. When 1 iteration is used in generating the UDBC codes (block length is 2), and 6 users access to the system, BER is

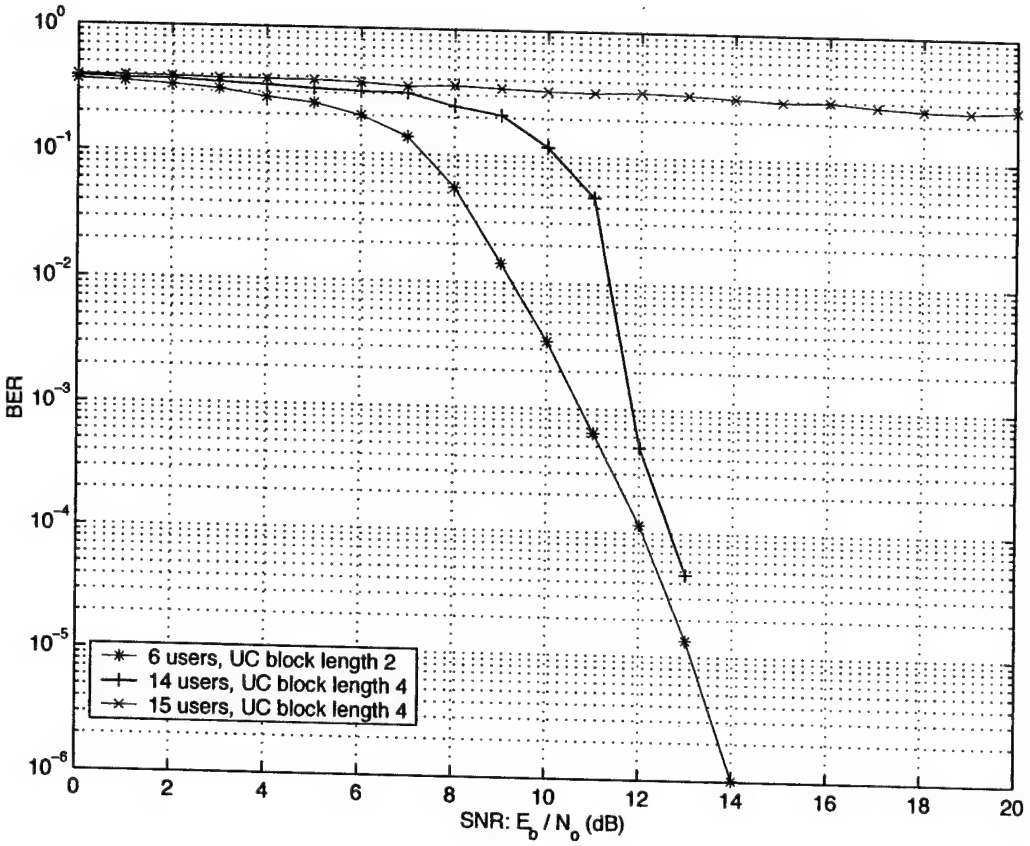

Figure 6.2: Performance of 3-frequency system, using rate 1/2 convolutional codes concatenated with uniquely decodable block codes.

below 10^{-5} when SNR is higher than 13dB. No error floor is observed for the performance. When 2 iterations are used for UDBC codes (block length is 4), 14 users access to the system, BER is around 10^{-5} when SNR is 13dB. Also, no error floor is observed. Such performances is decent for practical applications. However, for the 2 iteration case, maximum number of users is 16. When user number is larger than 14, the performance gets worse. As given in Figure 6.2, the error floor is extremely high (BER is around 10^{-1}) when 15 users access to the system. Also, it is easy to predict that when 16 users access to the system, high error floor will be observed either.

Generally, the maximum number of users need to access to the system so that the maximum sum rate is achieved. However, a large amount of the simulation results (not shown in Figure 6.2) display that most of the maximum sum rates given in Table 6.1 are not achievable, where high error floor can be observed. The reason for the high error floor has been discussed in Chapter 5. The error correction capability of the channel codes decides the level of error floor to a large extent, since the channel codes are extremely important in reducing the MUI among users. For our simulations, rate 1/2 RSC codes are applied, which is not strong enough to resolve the MUI problem when a large number of users access to the system. Thus, powerful error correction codes, which is suitable for resolving the serious MUI in the CFSK system, are

necessary to achieve higher code rate. Nonbinary codes such Reed-Solomon codes or convolutional codes over $GF(q)$ [40] are candidates for substituting the binary convolutional codes in former system.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

Iterative multiuser detection for CFSK system is investigated in this report. The theoretical background is provided in [17][18], which addressed the capacity issues for the synchronous CFSK channels. In Chapter 3, more capacity results for the asynchronous CFSK channels are discussed. The results show that the asynchronous CFSK channel perform similarly when the system load is low. When the system load gets higher, performance of the asynchronous CFSK channels gets worse. Thus, asynchronous CFSK system is suitable for application of moderate or low system load applications. For high system load applications, only synchronous CFSK system is usable. Results given in [18] show that the capacity of CFSK system is higher than the CDMA system. When the spreading factor is of moderate or high values,

the advantage of capacity for CFSK is the greatest. This is the reason that most of our research is focus on the applications for both synchronous and asynchronous CFSK system when spreading factor is moderate. We are also interested in the application of synchronous system when spreading factor is lower than 1. However, it is much more difficult in this region to achieve higher code rates than the bounds given by the multiuser CDMA system.

Techniques for multiuser detection is discussed in Chapter 4. The optimal detection results is provided by the MLSD approach. However, the high complexity of MLSD is impractical for applications. Iterative detection approaches are preferred. The multiuser channel metric given in Chapter 4 is of non-exponential complexity, which is extremely high when the number of users or frequencies is very large. Some simplification approaches is introduced, such as the KMPC and kMPF approaches. Results given in Chapter 5 show that the performance for KMPC and kMPF is decent when the spreading factor is high. When the system load get higher, the two approaches do not perform well since high MUI is introduced. To improve the performance of the simplified metric in the first iteration, NSMP approach is introduced in Chapter 4. Results display that the performance of NSMP in the first iteration approaches the performance of the optimal metric. The reduced MUI in the first iteration helps detection in later iterations. Thus, the BER

converges to a lower level when NSMP is applied in the first iteration.

The most important contribution in Chapter 5 is that the reason for the error floor is given. Basically, two factors lead to the error floor. First, the randomly generated RSC's are not guaranteed to be uniquely decodable. When spreading factor is high, the problem is not serious due to the low probability that two paths through the super-trellis have the same code sequence. In such circumstance, the randomly generated codes perform well. When spreading factor is close to 1 or lower than 1, the randomly generated codes will result in high error floor, since the codes in such condition are always not uniquely decodable. Note that this part of error floor exists for both MLSD and iterative detection.

The second factor for the error floor is the suboptimality of the iterative detection. In the iterative detection, though the users borrow the *a priori* information from each other during the iteration, there is no point in the consensus decoder that consider all the users as a whole. In this sense, the consensus decoder is just partly cooperative among the users. (The MLSD is a full cooperative approach.) Moreover, the first iteration will lead to poor estimation for the symbols, since no *apriori* information is available in the first iteration. When the imprecise *a priori* information is exchange to other users, MUI is introduced to other users. When the users'

codes are closely correlated with each other, the MUI becomes much more serious. If KMPC or kMPF is used for simplification, performance in the first iteration is deteriorated further. Heavy loaded system condition will also worsen the performance in the first iteration. The high MUI resulted from the first iteration will keep on affecting the performance of later iterations. The reduction of the MUI in each iteration depends on the error correction capability of the channel codes. If the channel codes are not strong enough in resolving the MUI, high level error floor is resulted.

From results in Chapter 5, we can conclude that to lower the error floor is effective in improving the system performance. To address the problem introduced by the non-uniquely decodable codes when spreading factor is low, the UDBC codes are introduce in Chapter 6. The results show that good performance can be achieved by using Chang and Wolf's non-binary block codes given in [17]. However, we find that the error floor is still extremely high if too many users access to the system. Thus, high coding rates cannot be achieved, which are approaching the capacity bound. The reason behind this is the suboptimality of iterative detection, since we find that the rate 1/2 RSC is not strong enough to reduce the MUI.

To solve problem caused by the the supoptimality of iterative detection, several approaches can be taken. At first, if KMPC or kMPF is used for

simplifying the metric, then NSMP should be used in the first iteration for reducing the MUI. Second, the channel interleaver can be used to reduce the correlation among the users' codes. Thus, the MUI caused in each iteration can be reduced. Results in Chapter 5 have shown the improvement by using interleavers. Since interleavers also increase the memory of the super-trellis code of the whole system, the performance given by interleavers is even better than the performance of MLSD without interleaver, though such comparison is not fair. Since the application of channel interleavers does not increase the complexity of the decoder structure, it is an effective approach for improve the system performance.

By comprehensively using the approaches discussed above, we achieved several high coding rates at the six points in Figure 7.1. Performance at point 5 and 6 are given in Chapter 6 by a synchronous system. At point 1, 2 and 3, both synchronous and asynchronous system can achieve decent performance. Good performance at point 4 can only be given by synchronous system. The details of these results can be found in Chapter 5. Note that all the six points are around or higher than the capacity of single-user CDMA. For the first four points, the rates are equal to or higher than the capacity of multiuser CDMA. This implies us that CFSK system has higher potential than CDMA system.

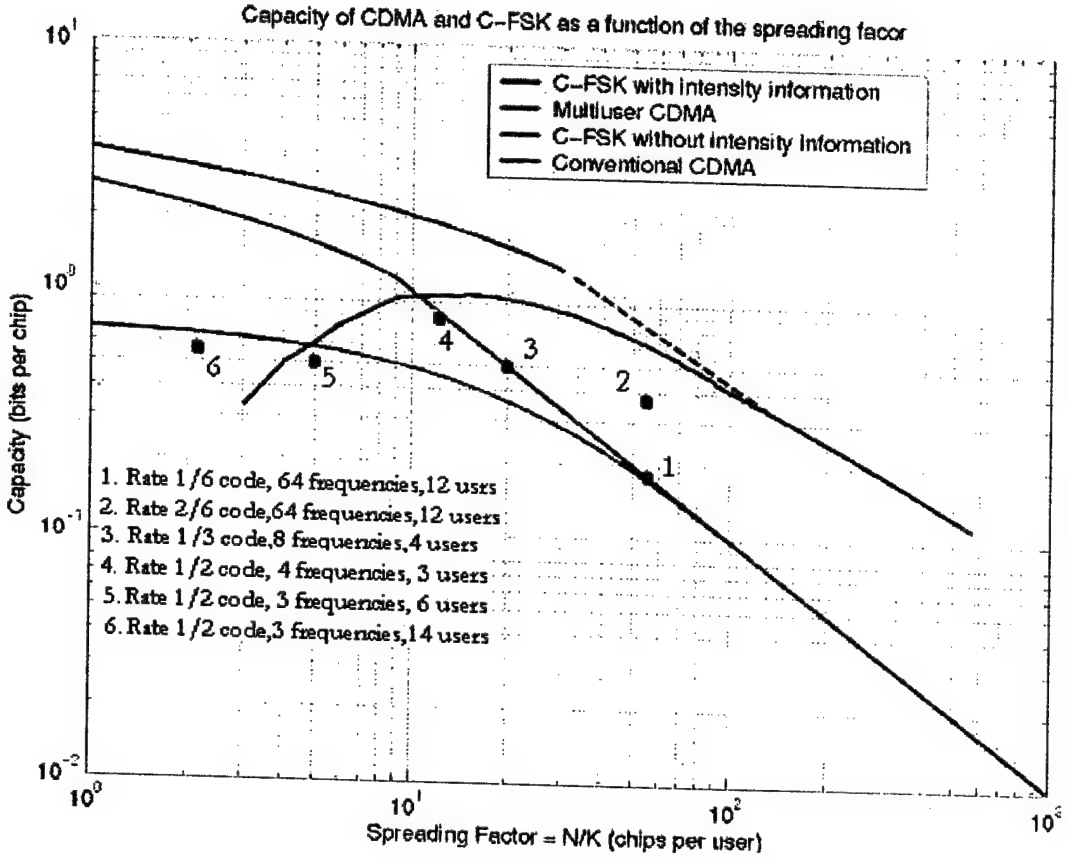

Figure 7.1: Achieved sum rates compared with capacity bounds.

7.2 Future Work

A problem caused by the signal mapping in our scheme is that it reduces the free distance of the convolutional codes. Performance of the convolutional codes is thus degraded and the codes are not strong enough to reduce the MUI when a large number of users access to the system. Possible approach to improve the performance of the users' component decoders is to map the bit sequence into the frequency sequence by non-binary codes with decent distance performance. Nonbinary codes, such as convolutional codes over $GF(q)$ [41][41][42][40] or convolution codes over rings [43][44][45][46][47], are candidates for substituting the binary convolutional codes in the former system. Using these nonbinary codes as building blocks, the more powerful nonbinary Turbo codes can be constructed [48][49]. The corresponding decoding algorithms [50][51] can be adapted for applications in CFSK system.

For the randomly generated nonbinary codes, unique decodability of these codes is still a major problem for applications of low spreading factors. To solve the problem, uniquely decodable block codes with large codeword numbers are useful. The KS code given by [37] and [39] provides us some inspiration for using the Reed-Solomon code to construct the uniquely decodable block code set. Also, researches of other nonbinary superimposed codes [37][38][39] also provide us clues for constructing the uniquely decodable block

code set.

We may also avoid generating the codes randomly. Instead, constructions for the nonbinary uniquely decodable trellis codes can be used. In this approach, we need to extend former results for the uniquely decodable trellis codes on the binary adder channel [26][52] to nonbinary uniquely decodable trellis codes on the CFSK channel with intensity information.

For future application of the CFSK system, performance of CFSK system on fading channels and the resistance of CFSK system to the near-far effects need to be studied in detail. Though the CFSK system is supposed to have decent performance under these channel conditions, a theoretical investigation is indeed necessary to prove the effectiveness of the CFSK system in dealing with these problems.

Bibliography

[1] J. G. Proakis and M. Salehi, *Communication Systems Engineering.* Upper Saddle River, NJ: Prentice Hall, 1994.

[2] F. European Telecommunications Standardization Institute, Sophia Antipolis, *Group Speciale Mobile or Globe System for Mobile Communication (GSM) Recommendation.*, 1988.

[3] W. D. Telecommunications Industry Association, TIA/EIA, *Mobile Station-Base Station Compatibility Standard for Dual-Mode Wideband Spread Spectrum Cellular System IS-95A.*, 1995.

[4] Qualcomm, *A Overview of The Application of Code Division Multiple Access (CDMA) to Digital cellular systems and personal cellular networks.* New Yark: Wiley, 1991.

[5] V. K. Garg, K. F. Smolik, and J. E. Wilkes, *Application of CDMA in Wireless/Personal Communications.* Prentice Hall, 1997.

[6] J. G. Proakis, *Digital Communications, Fourth Edition.* McGraw-Hill, 2001.

[7] S. Verdu, *Multiuser Detection.* Cambridge University Press, 1998.

[8] T. R. Giallorenzi and S. G. Wilson, "Multiuser ml sequence estimator for convolutionally coded asynchronous ds-cdma systems." *IEEE Trans. Comm.*, pp. 997–1008, August 1996.

[9] P. D. Alexander, L. K. Rasmussen, and C. B. Schlegel, "A linear receiver for coded multiuser cdma." *IEEE Trans. Comm.*, vol. COM-45, pp. 605–610, May 1997.

[10] M. C. Reed, C. B. Schlegel, P. D. Alexander, and J. A. Asenstorfer, "Iterative multiple detection for cdma with fec: near-single user performance." *IEEE Trans. Comm.*, pp. 1693–1699, December 1998.

[11] U. Timor, "Improved decoding scheme for frequency-hopped multilevel fsk system." *Bell Syst. Tech. J.*, pp. 1839–1855, December 1980.

[12] T. Mabuchi, R. Kohno, and H. Imai, "Multiuser detection scheme based on canceling cochannel interference for mfsk/fh-ssma system." *IEEE J. Select. Areas Commun.*, pp. 539–604, May 1994.

[13] N. Abramson, "Development of the alohanet." in *IEEE Trans. Inform. Theory*, March 1985, pp. 119–123.

[14] T. Aulin and R. Espineira, "Trellis coded multiple access (tcma)." in *Proc. Int. Conf. Commun. (ICC99)*, Vancouver, BC, Canada, June 1999, pp. 1177–1181.

[15] F. N. Brännström, "Trellis coded multiple access (tcma)-detectors and capacity considerations." Ph.D. dissertation, Chalmers University of Technology, Department of Computer Engineering, Göteborg, Sweden., November 2000.

[16] J. L. Massey and P. Mathys, "The collision channel without feedback." *IEEE Trans. Inform. Theory*, pp. 192–204, March 1985.

[17] S. Chang and J. K. Wolf, "On the t-user m-frequency noiseless multiple-access channel with and without intensity information." *IEEE Trans. Inform. Theory*, pp. 41–48, January 1981.

[18] A. J. Grant and C. Schlegel, "Collision-type multiple-user communications." *IEEE Trans. Inform. Theory*, vol. 43, pp. 1725–1736, September 1996.

[19] J. Hui, "Thoughput analysis for code division multiple accessing of the spread spectrum channel." *IEEE J. Select. Areas Commun.*, vol. SAC-2, pp. 482–486, July 1984.

[20] M. Rupf and J. L. Massey, "Optimum sequence multisets for synchronous code-division multiple-access channels." *IEEE Trans. Inform. Theory*, pp. 1261–1266, July 1994.

[21] A. J. Grant, "Multi-user information theory and coding." Ph.D. dissertation, University of South Australia, Moblie Communications Research Centre, Warrendi Rd. The Levels SA 5095, Australia, May 1996.

[22] D. M. Arnold, C. Schlegel, and L. C. Pérez, "Iterative multiuser detection of a multiple access fsk system." in *Proceedings of the 1997 IEEE Iternational Symposium on Information Theory*, Ulm, Germany, July 1997.

[23] F. N. Brännström, T. M. Aulin, and L. K. Rasmussen, "Iterative multiuser detection of trellis code multiple access using a posteriori probabilities." in *Proc. Int. Conf. Commun. (ICC2001)*, Helsinki, Finland, June 2001.

[24] T. Kasami and L. S, "Coding for a multiple-access channel." *IEEE Trans. Inform. Theory*, pp. 129–137, March 1976.

[25] R. Peterson and J. D. J. Costello, "Binary convolutional codes for a multiple-access channel." *IEEE Trans. Inform. Theory*, pp. 101–105, January 1979.

[26] P. R. Chevillat, "N-user trellis coding for a class of multiple-access channels." *IEEE Trans. Inform. Theory*, pp. 114–120, January 1981.

[27] S. Chang and J. E. J. Weldon, "Coding for t-user multiple-access channels." *IEEE Trans. Inform. Theory*, pp. 684–691, November 1979.

[28] P. Mathys, "A class of codes for a t active users out of n multiple-access communication system." *IEEE Trans. Inform. Theory*, pp. 1206–1219, November 1990.

[29] J. G. D. Forney, "Convolutional codes i: Algebraic structure." *IEEE Trans. Inform. Theory*, pp. 720–738, November 1970.

[30] R. Johannesson and K. S. Zigangirov, *Fundamentals of Convolutional Coding.* IEEE Press, New York, 1999.

[31] C. Xia and L. C. Pérez, "A cfsk system with iterative detection." in *Proc. IEEE GLOBECOM'04*, Dallas, Texas, USA, November 2004.

[32] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* New Yark: Wiley, 1991.

[33] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate." *IEEE Trans. Inform. Theory*, pp. 284–287, March 1974.

[34] P. Robertson, "Illuminating the structure of code and decoder of parallel concantenated recursive systematic (turbo) codes." in *Proc. CLOBECOM'94*, vol. 3, Dec. 1994, pp. 1298–1303.

[35] P. Robertson and T. Wörz, "A novel bandwidth efficient coding scheme employing turbo codes." in *Proc. ICC'96*, vol. 1, June 1996, pp. 962–967.

[36] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding: Turbo Codes." *Proc. 1993 IEEE International Conference on Communications, Geneva, Switzerland*, pp. 1064–1070, May 1993.

[37] W. H. Kautz and R. C. Singleton, "Nonrandom binary superimposed codes." *IEEE Trans. Inform. Theory*, pp. 363–377, April 1964.

[38] T. Ericson and V. I. Levenshtein, "Superimposed codes in the hamming space." *IEEE Trans. Inform. Theory*, pp. 1882–1893, November 1994.

[39] P. Z. Fan, M. Darnell, and B. Honary, "Superimposed codes for the multiaccess binary adder channel." *IEEE Trans. Inform. Theory*, pp. 1178–1182, July 1995.

[40] W. E. Ryan and S. G. Wilson, "Two classes of convolutional codes over $gf(q)$ for q-ary orthogonal signaling." *IEEE Trans. Comm.*, pp. 30–40, January 1991.

[41] J. P. Oenwalder, "Dual-k convolutional codes for noncoherently demodulated channels." in *Proc. Int. Telemetering Conf.*, vol. 12, 1976, pp. 165–174.

[42] I. F. Blake, "Properties of a class of nonbinary convolutional codes." in *Proc. 1987 conf. Inform. Sci. Syst.*, Johns Hopkins Univ., March 1987, p. 438.

[43] J. L. Massey and T. Mittelholzer, "Convolutional codes over rings." in *Proc. Joint Swedish-Soviet Int. Workshop on Inform. Theory*, Gotland, Sweeden, 1989, pp. 14–18.

[44] ——, "Systematicity and rotational invariance of convolutional codes over rings." in *Proc. Int. Workshop on Alg. and Combinatorial Coding Theory*, Leningrad, 1990, pp. 154–158.

[45] J. L. Massey, T. Mittelholzer, T. Riedel, and M. Vollenweider, "Ring convolutional codes for phas modulation." in *Proc. Int. Symp on Inform. Theory*, San Diego, CA, 1990.

[46] T. Mittelholzer, "Minimal encoders for convolutional codes over rings." in *Communication Theory and Applications: Systems, Signal Processing and Error Control Coding*, HW Comm. Ltd., 1993, pp. 30–36.

[47] ——, "Convolutional codes over rings and the two chain conditions." in *Proc. Int. Symp. on Inform. and Coding Theory*, Ulm, Germany, 1997, p. 285.

[48] C. Berrou and M. Jézéquel, "Non-binary convolutional codes for turbo coding." *Electronics Letters*, pp. 39–40, 1993.

[49] C. Berrou, M. Jézéquel, C. Douillard, and S. Kerouédan, "The advantages of non-binary turbo codes." in *Proc. ITW2001*, Cairns, Australia, September 2001.

[50] J. Berkmann, "On turbo decoding of nonbinary codes." *IEEE Comm. Letters*, pp. 94–96, 1998.

[51] A. J. Grant, "Convergence of non-binary iterative decoding." in *Proc. IEEE GLOBECOM'01*, San Antonio, Texas, USA, November 2001, pp. 1058–1062.

[52] G. H. Khachatrian and S. S. Martirossian, "Code consruction for the t-user noiseless adder channel." *IEEE Trans. on Inform. Theory*, vol. 44, pp. 1953–1957, September 1998.